国外电子与通信教材系列

现代通信系统

（MATLAB版）（第二版）

Contemporary Communication Systems Using MATLAB and Simulink

Second Edition



John G. Proakis
［美］ Masoud Salehi 著
Gerhard Bauch
刘树棠 译

THOMSON



電子工業出版社
Publishing House of Electronics Industry
http://www.phei.com.cn

现代通信系统

（MATLAB版）（第二版）

Contemporary Communication Systems Using MATLAB and Simulink

Second Edition

利用MATLAB拓展学习空间

本书采用MATLAB作为“虚拟实验室”，给出了解决“现代通信系统”课程中涉及的各方面问题的分析思路、方法、MATLAB脚本（或程序）文件和处理结果示例，还给出了许多可供学生自主学习和研讨的习题以及作业。全书的内容覆盖了信号与线性系统、随机过程、模拟调制、模拟－数字转换、基带数字传输、带限信道的数字传输、载波调制的数字传输、信道容量和编码、扩频通信系统以及数字调制方法的Simulink仿真等。

本书特点

- 弱化推导证明，注重步骤，强调内容的系统性并注重相互联系
- 以“解说题”等形式来演绎和深化概念，并在重要概念的深度和广度上进行适当延伸
- 丰富的MATLAB脚本和可随时利用的网站为读者提供了充分的资源

网站资源

- 本书的Web站点http://engineering.brookscole.com/ccsystems2e包括了教材中用到的所有MATLAB和Simulink文件

作者简介

John G. Proakis：美国东北大学著名教授，长期从事数字通信和数字信号处理等方面的教学和研究，主要著作有《数字通信（第四版）》、《通信系统工程》、《高级数字信号处理》和《通信系统原理》等。





THOMSON



责任编辑：马　岚

封面设计：毛惠庚



ISBN 7-121-01042-9　　定价：36.00元

国外电子与通信教材系列

现代通信系统

（MATLAB版）（第二版）

Contemporary Communication Systems Using MATLAB and Simulink

Second Edition

John G. Proakis
［美］ Masoud Salehi 著
Gerhard Bauch

刘树棠 译

電子工業出版社
Publishing House of Electronics Industry
北京 · BEIJING

内容简介

本书提供了利用MATLAB在计算机上解决"现代通信系统"课程中涉及的各方面问题的分析思路、方法、MATLAB脚本(或程序)文件和处理结果示例，还给出了许多可供学生自主学习和研讨的习题以及作业。全书内容共10章，分别讨论了信号与线性系统、随机过程、模拟调制、模拟－数字转换、基带数字传输、带限信道的数字传输、载波调制的数字传输、信道容量和编码、扩频通信系统以及数字调制方法的Simulink仿真等。

本书适合已具备MATLAB基本知识的通信工程、电子工程、电气工程、计算机工程和计算机科学等专业方向的高年级本科生和研究生作为相关课程的参考书和补充教材，也可供有关的教师和工程技术人员参考使用。

981-265-398-8

版权贸易合同登记号　图字：01-2003-1229

图书在版编目(CIP)数据

现代通信系统(MATLAB版)(第二版)/(美)普罗克斯(Proakis, J. G.)等著；刘树棠译.
北京：电子工业出版社，2005.4
(国外电子与通信教材系列)
书名原文：Contemporary Communication Systems: Using MATLAB and Simulink, Second Edition
ISBN 7-121-01042-9

Ⅰ.现… Ⅱ.①普… ②刘… Ⅲ.通信系统－计算机辅助计算－软件包，MATLAB－教材 Ⅳ.TN914

中国版本图书馆CIP数据核字(2005)第022800号

责任编辑：马　岚
印　　刷：北京顺义兴华印刷厂
出版发行：电子工业出版社
　　　　　北京市海淀区万寿路173信箱　邮编：100036
经　　销：各地新华书店
开　　本：787 × 1092　1/16　　印张：25　字数：640千字
印　　次：2005年4月第1次印刷
定　　价：36.00元

凡购买电子工业出版社的图书，如有缺损问题，请向购买书店调换；若书店售缺，请与本社发行部联系。联系电话：(010)68279077。质量投诉请发邮件至zlts@phei.com.cn，盗版侵权举报请发邮件至dbqq@phei.com.cn。

序

2001年7月间，电子工业出版社的领导同志邀请各高校十几位通信领域方面的老师，商量引进国外教材问题。与会同志对出版社提出的计划十分赞同，大家认为，这对我国通信事业、特别是对高等院校通信学科的教学工作会很有好处。

教材建设是高校教学建设的主要内容之一。编写、出版一本好的教材，意味着开设了一门好的课程，甚至可能预示着一个崭新学科的诞生。20世纪40年代MIT林肯实验室出版的一套28本雷达丛书，对近代电子学科、特别是对雷达技术的推动作用，就是一个很好的例子。

我国领导部门对教材建设一直非常重视。20世纪80年代，在原教委教材编审委员会的领导下，汇集了高等院校几百位富有教学经验的专家，编写、出版了一大批教材；很多院校还根据学校的特点和需要，陆续编写了大量的讲义和参考书。这些教材对高校的教学工作发挥了极好的作用。近年来，随着教学改革不断深入和科学技术的飞速进步，有的教材内容已比较陈旧、落后，难以适应教学的要求，特别是在电子学和通信技术发展神速、可以讲是日新月异的今天，如何适应这种情况，更是一个必须认真考虑的问题。解决这个问题，除了依靠高校的老师和专家撰写新的符合要求的教科书外，引进和出版一些国外优秀电子与通信教材，尤其是有选择地引进一批英文原版教材，是会有好处的。

一年多来，电子工业出版社为此做了很多工作。他们成立了一个"国外电子与通信教材系列"项目组，选派了富有经验的业务骨干负责有关工作，收集了230余种通信教材和参考书的详细资料，调来了100余种原版教材样书，依靠由20余位专家组成的出版委员会，从中精选了40多种，内容丰富，覆盖了电路理论与应用、信号与系统、数字信号处理、微电子、通信系统、电磁场与微波等方面，既可作为通信专业本科生和研究生的教学用书，也可作为有关专业人员的参考材料。此外，这批教材，有的翻译为中文，还有部分教材直接影印出版，以供教师用英语直接授课。希望这些教材的引进和出版对高校通信教学和教材改革能起一定作用。

在这里，我还要感谢参加工作的各位教授、专家、老师与参加翻译、编辑和出版的同志们。各位专家认真负责、严谨细致、不辞辛劳、不怕琐碎和精益求精的态度，充分体现了中国教育工作者和出版工作者的良好美德。

随着我国经济建设的发展和科学技术的不断进步，对高校教学工作会不断提出新的要求和希望。我想，无论如何，要做好引进国外教材的工作，一定要联系我国的实际。教材和学术专著不同，既要注意科学性、学术性，也要重视可读性，要深入浅出，便于读者自学；引进的教材要适应高校教学改革的需要，针对目前一些教材内容较为陈旧的问题，有目的地引进一些先进的和正在发展中的交叉学科的参考书；要与国内出版的教材相配套，安排好出版英文原版教材和翻译教材的比例。我们努力使这套教材能尽量满足上述要求，希望它们能放在学生们的课桌上，发挥一定的作用。

最后，预祝"国外电子与通信教材系列"项目取得成功，为我国电子与通信教学和通信产业的发展培土施肥。也恳切希望读者能对这些书籍的不足之处、特别是翻译中存在的问题，提出意见和建议，以便再版时更正。

吴佑寿

中国工程院院士、清华大学教授

"国外电子与通信教材系列"出版委员会主任

出 版 说 明

进入21世纪以来，我国信息产业在生产和科研方面都大大加快了发展速度，并已成为国民经济发展的支柱产业之一。但是，与世界上其他信息产业发达的国家相比，我国在技术开发、教育培训等方面都还存在着较大的差距。特别是在加入WTO后的今天，我国信息产业面临着国外竞争对手的严峻挑战。

作为我国信息产业的专业科技出版社，我们始终关注着全球电子信息技术的发展方向，始终把引进国外优秀电子与通信信息技术教材和专业书籍放在我们工作的重要位置上。在2000年至2001年间，我社先后从世界著名出版公司引进出版了40余种教材，形成了一套“国外计算机科学教材系列”，在全国高校以及科研部门中受到了欢迎和好评，得到了计算机领域的广大教师与科研工作者的充分肯定。

引进和出版一些国外优秀电子与通信教材，尤其是有选择地引进一批英文原版教材，将有助于我国信息产业培养具有国际竞争能力的技术人才，也将有助于我国国内在电子与通信教学工作中掌握和跟踪国际发展水平。根据国内信息产业的现状、教育部《关于“十五”期间普通高等教育教材建设与改革的意见》的指示精神以及高等院校老师们反映的各种意见，我们决定引进“国外电子与通信教材系列”，并随后开展了大量准备工作。此次引进的国外电子与通信教材均来自国际著名出版商，其中影印教材约占一半。教材内容涉及的学科方向包括电路理论与应用、信号与系统、数字信号处理、微电子、通信系统、电磁场与微波等，其中既有本科专业课程教材，也有研究生课程教材，以适应不同院系、不同专业、不同层次的师生对教材的需求，广大师生可自由选择和自由组合使用。我们还将与国外出版商一起，陆续推出一些教材的教学支持资料，为授课教师提供帮助。

此外，“国外电子与通信教材系列”的引进和出版工作得到了教育部高等教育司的大力支持和帮助，其中的部分引进教材已通过“教育部高等学校电子信息科学与工程类专业教学指导委员会”的审核，并得到教育部高等教育司的批准，纳入了“教育部高等教育司推荐——国外优秀信息科学与技术系列教学用书”。

为做好该系列教材的翻译工作，我们聘请了清华大学、北京大学、北京邮电大学、东南大学、西安交通大学、天津大学、西安电子科技大学、电子科技大学等著名高校的教授和骨干教师参与教材的翻译和审校工作。许多教授在国内电子与通信专业领域享有较高的声望，具有丰富的教学经验，他们的渊博学识从根本上保证了教材的翻译质量和专业学术方面的严格与准确。我们在此对他们的辛勤工作与贡献表示衷心的感谢。此外，对于编辑的选择，我们达到了专业对口；对于从英文原书中发现的错误，我们通过与作者联络、从网上下载勘误表等方式，逐一进行了修订；同时，我们对审校、排版、印制质量进行了严格把关。

今后，我们将进一步加强同各高校教师的密切关系，努力引进更多的国外优秀教材和教学参考书，为我国电子与通信教材达到世界先进水平而努力。由于我们对国内外电子与通信教育的发展仍存在一些认识上的不足，在选题、翻译、出版等方面的工作中还有许多需要改进的地方，恳请广大师生和读者提出批评及建议。

电子工业出版社

教材出版委员会

译　者　序

本书的第二版与前一版本相比，主要的区别是补充了如下内容：数字通信系统的 Monte Carlo 仿真(见 2.7 节)、矢量量化(见 4.3.2 节)、多载波调制和 OFDM(见 7.6 节)以及第 10 章。因此，原来的个别章节在次序和内容方面也做了相应的改变，并增加了相应的习题和例题。

本书作者的意图大致有两个方面：一方面是探索如何将基于计算机的学习工具融于教学；另一方面是给学生提供额外的学习空间，让他们能够自主学习。总的期望是让学生积极主动地学习，因为这样会学得更好。因此，本书在写作上大致有如下特点：

- 不求过细，而是强调内容的系统性并注重相互联系。本书的内容提纲挈领，注意总结，联系实际，因此一般来说适合作为辅助性教材。
- 用“解说题”等形式来演绎和深化概念，并在重要概念的深度和广度上进行适当延伸。
- 丰富的 MATLAB 脚本资源和可随时利用的网站为广大读者提供了很大的学习空间，使他们的主要精力不再耗费在编程上，而能将精力集中在探究上。读者可以利用这些脚本资源做自己想做的事，通过计算机仿真得到初步训练并获得更多领悟，使自己的创新意识和才能发挥得到无限拓展空间。

译者认为，对待本书与其说“学”不如说“做”更为贴切，即按照本书所列的解说题和习题认真地在计算机上去“做”，尤其对于第 10 章所涵盖的那些作业更是如此。

在本书的翻译过程中，虽然有第一版作为基础，有些内容可以借鉴，但我还是逐字逐句地进行了斟酌，对其中不少不够贴切或有错误的地方进行了勘正。即使如此，我仍感到有个别地方不太有把握，欢迎读者批评指正。

最后对我的老伴孙漪教授为这个中译本的完成所付出的艰辛和支持表示衷心的感谢，对电子工业出版社给予的信任和帮助致诚挚的谢意。

刘树棠
于西安交通大学

前　言

在当今市场上，很多教材都在讨论模拟和数字通信系统中的基本论题，其中包括编码和解码算法、调制和解调技术等。根据需要，这些教材中的大多数都重点关注构成各种模块（例如编码器、解码器、调制器和解调器等）的设计和性能分析的基本理论，而这些模块又是组成一个通信系统的基本单元。然而，只有相当少的教材（尤其是为大学本科生而写的教材）包含激励学生学习的各种各样的应用例子。

本书的意图

本书的目的是用做任何一本在通信系统方面较全面的教材的参考书或补充教材。书中提供了种类繁多的大量练习，读者可以利用 MATLAB 的普及型学生版在计算机（普通 PC 机即可）上解题。本书主要是为电气工程、计算机工程和计算机科学系的高年级本科生以及研究生准备的，认为这些学生（或用户）已经熟悉了 MATLAB 的基本知识。由于在 MATLAB 方面已有多种辅导书籍或指南手册，所以这些专题未包括在本书内。

从设计方面来讲，各种论题的处理都很简明扼要。对于每个论题，书中给出了讨论的初衷和简短的叙述，建立了必要的符号，然后用例子阐明基本概念。我们期望学生所用的基本教材和授课教师都已对所涉及的论题提供了必要的理论深度。例如，在介绍匹配滤波器和相关器时，就要讲到这些器件可以使被加性高斯白噪声（AWGN）污损的信号得到最佳解调。但是，对此我们并不给出证明，在大多数有关通信系统方面的教材中一般都会给出这样的证明。

本书的组成

本书共分 10 章。前两章是有关信号与线性系统和随机过程方面的内容，给出了在通信系统学习中所要求的基本背景内容。第 3 章是关于模拟通信技术的，第 4 章讨论了模拟 – 数字转换内容，接下来的 5 章都重点讨论的是数字通信。第 10 章讲述了 Simulink 仿真。

第 1 章　信号与线性系统

第 1 章回顾了线性系统分析中的一些基本方法和技术，其中既包括时域特性，也包括频域特性。由于频域分析技术在通信系统分析和处理中使用得最多，所以我们的讨论重点放在频域分析上。

第 2 章　随机过程

第 2 章讨论了产生随机变量和随机过程样本的方法，包括具有给定概率分布函数的随机变量的产生，高斯和高斯–马尔可夫过程样本的产生，以及平稳随机过程在时域和频域中的特性。这一章还讨论了经由 Monte Carlo 仿真的概率估计问题。

第 3 章　模拟调制

第 3 章讨论的是在有加性噪声和无加性噪声时，模拟调制和解调技术的性能。被研究的

系统包括幅度调制(AM)(例如双边带 AM,单边带 AM 和常用的 AM)和角调制(如频率调制 FM 和相位调制 PM)。

第 4 章　模拟 - 数字转换

第 4 章研究的是将模拟源信号有效地转换为数字序列的各种方法,转换的结果是可以用数字方式传输或存储信号。既要考虑有损数据压缩方法(如脉冲编码调制 PCM),也要讨论无损数据压缩方法(如 Huffman 编码)。

第 5 章　基带数字传输

第 5 章讨论了经由 AWGN 信道传输数字信息的基带数字调制和解调技术。二进制和非二进制的调制技术都要考虑。要讨论这些信号的最佳解调,并对这类解调器的性能进行评价。

第 6 章　带限信道的数字传输

第 6 章讨论的是带限信道的特性和针对此类信道的信号波形设计问题。我们说明了信道失真会导致码间干扰(ISI),这会引起在信号解调中的差错。然后,我们将处理有关补偿信道失真的信道均衡器的设计问题。

第 7 章　载波调制的数字传输

第 7 章讨论了 4 种类型的载波调制信号,它们都适合在带通信道内进行传输,分别是幅度调制信号、正交幅度调制信号、相移键控和频移键控。第 7 章还讨论了正交频分多路复用(OFDM)。

第 8 章　信道容量和编码

第 8 章对通信信道建立了一个适当的数学模型,并介绍了一个称为“信道容量”的基本量,它给出了这条信道能够传输的信息量的限制。本章讨论了两种信道模型:二进制对称信道(BSC)和加性高斯白噪声(AWGN)信道。在处理分组和卷积码以实现经由这些信道的可靠通信中,都将使用这些信道模型。

第 9 章　扩频通信系统

第 9 章要讨论扩频数字通信系统的基本原理。特别是直接序列扩频(简称直扩,DS)系统和跳频(FH)系统将分别与相移键控(PSK)和频移键控(FSK)调制一起考虑。本章还要讨论在扩频系统中要用到的伪随机噪声(PN)序列的产生。

第 10 章　数字调制方法的 Simulink 仿真指南

第 10 章介绍了 Simulink 及其在数字调制系统仿真中的应用。我们首先对 Simulink 进行了简单介绍,基本上涵盖了系统仿真的基础。接下来的几节给出了各种数字通信系统仿真的许多例子。

网址

本书的 Web 站点(http://engineering.brookscole.com/ccsystems2e)包括了教材中用到的所有 MATLAB 和 Simulink 文件。文件是对应于各章以单独的目录给出的。有些 MATLAB 文件出现在几个目录中,这是由于它们应用在几章中。大多数文件都加有许多注释,以使这些文件更容易理解。然而,在开发这些文件时,我们的主要目的在于代码的清晰和可读性,而不在于它的高效性。当使用最高效的代码而造成文件不易阅读和理解时,就应该选取一些更易阅读而稍欠高效的代码。

为了使用这些文件,读者要将全部目录复制到计算机中,并将相应的路径添加到 matlabpath 环境中。在 IBM 兼容机中,通常可通过编辑 matlabrc.m 文件来完成。所有文件都使用 MATLAB 6 进行了测试。

致谢

本书的 Simulink 仿真指南是在德国慕尼黑科技大学通信工程学院开发的实验课程的基础上进行修改和扩充的版本。我们谨对 Joachim Hagenauer 教授支持写作本书并允许使用这个软件表示感谢。同时,我们也要感谢 Christian Buchner 和 Christoph Renner,他们完成了大部分编程工作。我们对 MathWorks 允许提供标准学生版中未包括的某些 Simulink 模块表示衷心感谢,还要对 MathWorks 的 Stuart McGarrity 所给予的有益指点表示诚挚的谢意。

本书经过了 Hyuck M. Kwon(Wichita 州立大学),Ting-Chung Poon(弗吉尼亚理工学院和州立大学)和 Sumit Roy(华盛顿大学)的审阅,对于他们给出的很有见地的评注,我们表示由衷的感谢。

目　　录

第1章 信号与线性系统

1.1 概述

本章回顾了在通信系统分析中用到的一些线性系统分析的基本方法和技术。在通信系统研究中有两个基本问题是一定要搞清楚的，其中一个问题是线性系统及其在时域和频域的特性，另一个问题是随机信号的概率与分析。大多数通信信道以及发射和接收装置中的很多部分，都可以用线性时不变(LTI)系统来建模，所以来自于线性系统分析中的一些著名方法和技术都能在通信系统分析中使用。我们的讨论重点放在频域分析方法上，因为在通信系统分析中这是最常用的技术。本章的讨论从傅里叶级数和傅里叶变换开始，然后再包括功率和能量的概念、采样定理以及带通信号的低通表示等。

1.2 傅里叶级数

一个线性时不变系统的输入/输出关系由如下卷积积分定义：

$$y(t)=x(t)*h(t)=\int_{-\infty}^{\infty}h(\tau)x(t-\tau)\mathrm{d}\tau \tag{1.2.1}$$

式中 $h(t)$记为系统的冲激响应，$x(t)$为输入信号，$y(t)$则为输出信号。如果 $x(t)$是一个由

$$x(t)=A\mathrm{e}^{\mathrm{j}2\pi f_0 t} \tag{1.2.2}$$

给出的复指数信号，那么输出则为：

$$\begin{aligned}y(t)&=\int_{-\infty}^{\infty}A\mathrm{e}^{\mathrm{j}2\pi f_0(t-\tau)}h(\tau)\mathrm{d}\tau\\&=A\left[\int_{-\infty}^{\infty}h(\tau)\mathrm{e}^{-\mathrm{j}2\pi f_0\tau}\mathrm{d}\tau\right]\mathrm{e}^{\mathrm{j}2\pi f_0 t}\end{aligned} \tag{1.2.3}$$

换句话说，该输出是一个与输入信号具有相同频率的复指数。但是，输出的(复)振幅是输入(复)振幅乘以：

$$\int_{-\infty}^{\infty}h(\tau)\mathrm{e}^{-\mathrm{j}2\pi f_0\tau}\mathrm{d}\tau$$

应该注意到，上式这个量是该 LTI 系统冲激响应 $h(t)$和输入信号频率 f_0 的函数。因此，计算 LTI 系统对指数输入的响应特别容易。这样，在线性系统分析中应该寻找一些将信号展开成复指数之和的方法。**傅里叶级数和傅里叶变换就是利用复指数来展开信号的技术。**

当采用信号集合$\{\mathrm{e}^{\mathrm{j}2\pi nt/T_0}\}_{n=-\infty}^{\infty}$作为展开式的基时，傅里叶级数就是周期为 T_0 的周期信号的正交展开。利用这个基，任何周期为 T_0 的周期信号 $x(t)$① 都可以表示为：

---

① 傅里叶级数存在的充分条件是 $x(t)$满足狄里赫利(Dirichlet)条件。详见参考文献[1]。

$$x(t) = \sum_{n=-\infty}^{\infty} x_n e^{j2\pi nt/T_0} \tag{1.2.4}$$

式中 x_n 称为该信号 $x(t)$ 的**傅里叶级数系数**，并由下式给出：

$$x_n = \frac{1}{T_0}\int_{\alpha}^{\alpha+T_0} x(t)e^{-j2\pi nt/T_0}\,dt \tag{1.2.5}$$

其中，α 是任意常数，可按使该积分式计算简便来选取。频率 $f_0 = 1/T_0$ 称为该周期信号的**基波频率**，而 $f_n = nf_0$ 称为**第 n 次谐波**。在大多数情况下，$\alpha = 0$ 或 $\alpha = -T_0/2$ 都是一种好的选取方案。

这种形式的傅里叶级数称为**指数形式的傅里叶级数**，它既能用于实值，也能用于复值信号 $x(t)$，只要它们都是周期的。一般来说，傅里叶级数 $\{x_n\}$ 是复数，即使当 $x(t)$ 是一个实值信号时也是如此。

当 $x(t)$ 是一个**实值**周期信号时，有：

$$\begin{aligned} x_{-n} &= \frac{1}{T_0}\int_{\alpha}^{\alpha+T_0} x(t)e^{j2\pi nt/T_0}\,dt \\ &= \frac{1}{T_0}\left[\int_{\alpha}^{\alpha+T_0} x(t)e^{-j2\pi nt/T_0}\,dt\right]^* \\ &= x_n^* \end{aligned} \tag{1.2.6}$$

由此显然可得：

$$\begin{cases} |x_n| = |x_{-n}| \\ \angle x_n = -\angle x_{-n} \end{cases} \tag{1.2.7}$$

因此，实值信号的傅里叶级数系数具有**埃尔米特对称性**，即它们的幅值是偶函数，相位是奇函数(或者说，它们的实部是偶函数，虚部是奇函数)。

傅里叶级数的另一种形式称为**三角函数形式的傅里叶级数**，它仅适用于实值周期信号。定义

$$x_n = \frac{a_n - jb_n}{2} \tag{1.2.8}$$

$$x_{-n} = \frac{a_n + jb_n}{2} \tag{1.2.9}$$

再利用欧拉公式

$$e^{-j2\pi nt/T_0} = \cos\left(2\pi t\frac{n}{T_0}\right) - j\sin\left(2\pi t\frac{n}{T_0}\right) \tag{1.2.10}$$

可得：

$$\begin{aligned} a_n &= \frac{2}{T_0}\int_{\alpha}^{\alpha+T_0} x(t)\cos\left(2\pi t\frac{n}{T_0}\right)dt \\ b_n &= \frac{2}{T_0}\int_{\alpha}^{\alpha+T_0} x(t)\sin\left(2\pi t\frac{n}{T_0}\right)dt \end{aligned} \tag{1.2.11}$$

因此，

$$x(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n\cos\left(2\pi t\frac{n}{T_0}\right) + b_n\sin\left(2\pi t\frac{n}{T_0}\right) \tag{1.2.12}$$

注意，对于 $n = 0$，总有 $b_0 = 0$，所以 $a_0 = 2x_0$，再定义

$$\begin{cases} c_n = \sqrt{a_n^2 + b_n^2} \\ \theta_n = -\arctan \dfrac{b_n}{a_n} \end{cases} \tag{1.2.13}$$

并利用下列关系：

$$a\cos\phi + b\sin\phi = \sqrt{a^2 + b^2}\cos\left(\phi - \arctan\frac{b}{a}\right) \tag{1.2.14}$$

就可以将式(1.2.12)写成

$$x(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} c_n \cos\left(2\pi t \frac{n}{T_0} + \theta_n\right) \tag{1.2.15}$$

这就是对于实值周期信号傅里叶级数展开的第三种形式。实值周期信号的傅里叶级数$\{x_n\}$与a_n, b_n, c_n, θ_n 的关系一般是通过下式联系起来的：

$$\begin{cases} a_n = 2\mathrm{Re}[x_n] \\ b_n = -2\mathrm{Im}[x_n] \\ c_n = |x_n| \\ \theta_n = \angle x_n \end{cases} \tag{1.2.16}$$

将$|x_n|$和$\angle x_n$ 对 n 或 nf_0 作图，该图称为 $x(t)$的**离散频谱**。$|x_n|$的图通常称为**幅度谱**，而$\angle x_n$ 的图则称为**相位谱**。

如果 $x(t)$是实值且为偶函数，即 $x(-t) = x(t)$，那么取 $\alpha = -T_0/2$ 之后则有

$$b_n = \frac{2}{T_0}\int_{-T_0/2}^{T_0/2} x(t)\sin\left(2\pi t \frac{n}{T_0}\right)\mathrm{d}t \tag{1.2.17}$$

因为被积函数是 t 的奇函数，该积分的值为零。因此，对于实值偶信号 $x(t)$来说，所有 x_n 都是实数。这时三角函数型的傅里叶级数全部由余弦函数构成。同理，若 $x(t)$是实值且为奇函数，即 $x(-t) = -x(t)$，则有

$$a_n = \frac{2}{T_0}\int_{\alpha}^{\alpha+T_0} x(t)\cos\left(2\pi t \frac{n}{T_0}\right)\mathrm{d}t \tag{1.2.18}$$

的积分值为零，所有 x_n 均为虚数，这时三角函数型傅里叶级数全部由正弦函数构成。

解说题

解说题 1.1　[矩形信号串的傅里叶级数]

设周期为 T_0 的周期信号 $x(t)$定义为：

$$x(t) = A\Pi\left(\frac{t}{2t_0}\right) = \begin{cases} A, & |t| < t_0 \\ \dfrac{A}{2}, & t = \pm t_0 \\ 0, & \text{其余 } t \end{cases} \tag{1.2.19}$$

这里，$|t| \leqslant T_0/2$，$t_0 < T_0/2$。矩形信号 $\Pi(t)$按惯例定义为：

$$\Pi(t) = \begin{cases} 1, & |t| < \dfrac{1}{2} \\ \dfrac{1}{2}, & t = \pm\dfrac{1}{2} \\ 0, & \text{其余 } t \end{cases} \tag{1.2.20}$$

图 1.1 所示的是 $x(t)$ 的图。

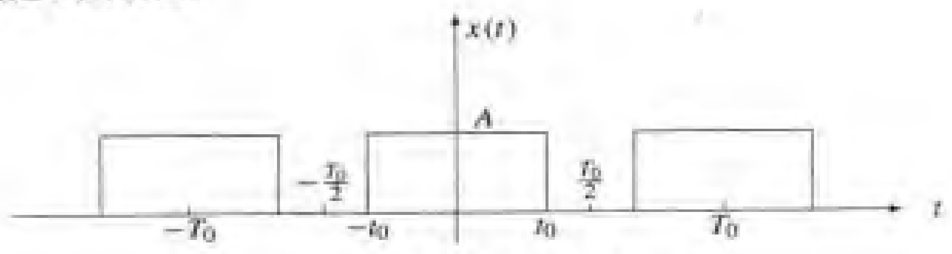


图 1.1 解说题 1.1 中的信号 $x(t)$

假定 $A=1, T_0=4$ 和 $t_0=1$。

1. 求分别用指数和三角函数展开的 $x(t)$ 的傅里叶级数系数。
2. 画出 $x(t)$ 的离散谱。

题 解

1. 导出 $x(t)$ 展开式的傅里叶级数系数有：

$$x_n = \frac{1}{4}\int_{-1}^{1} e^{-j2\pi nt/4}dt$$

$$= \frac{1}{-2j\pi n}[e^{-j2\pi n/4} - e^{j2\pi n/4}] \tag{1.2.21}$$

$$= \frac{1}{2}\text{sinc}\left(\frac{n}{2}\right) \tag{1.2.22}$$

其中 $\text{sinc}(x)$ 定义为：

$$\text{sinc}(x) = \frac{\sin(\pi x)}{\pi x} \tag{1.2.23}$$

图 1.2 就是 sinc 函数的图形。显然，所有 x_n 都是实数，因为 $x(t)$ 为实函数且为偶函数，所以

$$\begin{cases} a_n = \text{sinc}\left(\frac{n}{2}\right) \\ b_n = 0 \\ c_n = \left|\text{sinc}\left(\frac{n}{2}\right)\right| \\ \theta_n = 0, \pi \end{cases} \tag{1.2.24}$$

注意，当 n 为偶数时，$x_n=0\left(n=0\text{ 例外，这时 } a_0=c_0=1\text{，并且 } x_0=\frac{1}{2}\right)$。利用这些系数，有

$$x(t) = \sum_{n=-\infty}^{\infty}\frac{1}{2}\text{sinc}\left(\frac{n}{2}\right)e^{j2\pi nt/4}$$

$$= \frac{1}{2} + \sum_{n=1}^{\infty}\text{sinc}\left(\frac{n}{2}\right)\cos\left(2\pi t\frac{n}{4}\right) \tag{1.2.25}$$

图 1.3 显示的是当 $n=0, 1, 3, 5, 7$ 和 9 时，该傅里叶级数在一个周期内对该信号的近似情况。可以注意到，随着 n 的增加，近似的波形会变得越来越接近原信号 $x(t)$。

2. 注意，x_n 总是实数，因此根据它的符号，相位不是 0 就是 π。x_n 的幅度是 $\frac{1}{2}\left|\text{sinc}\left(\frac{n}{2}\right)\right|$，离散谱如图 1.4 所示。

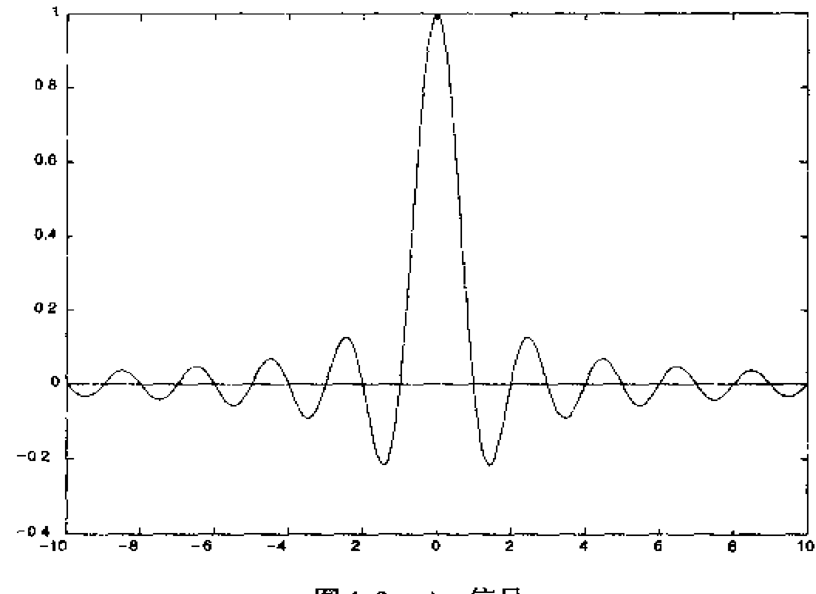


图 1.2　sinc 信号

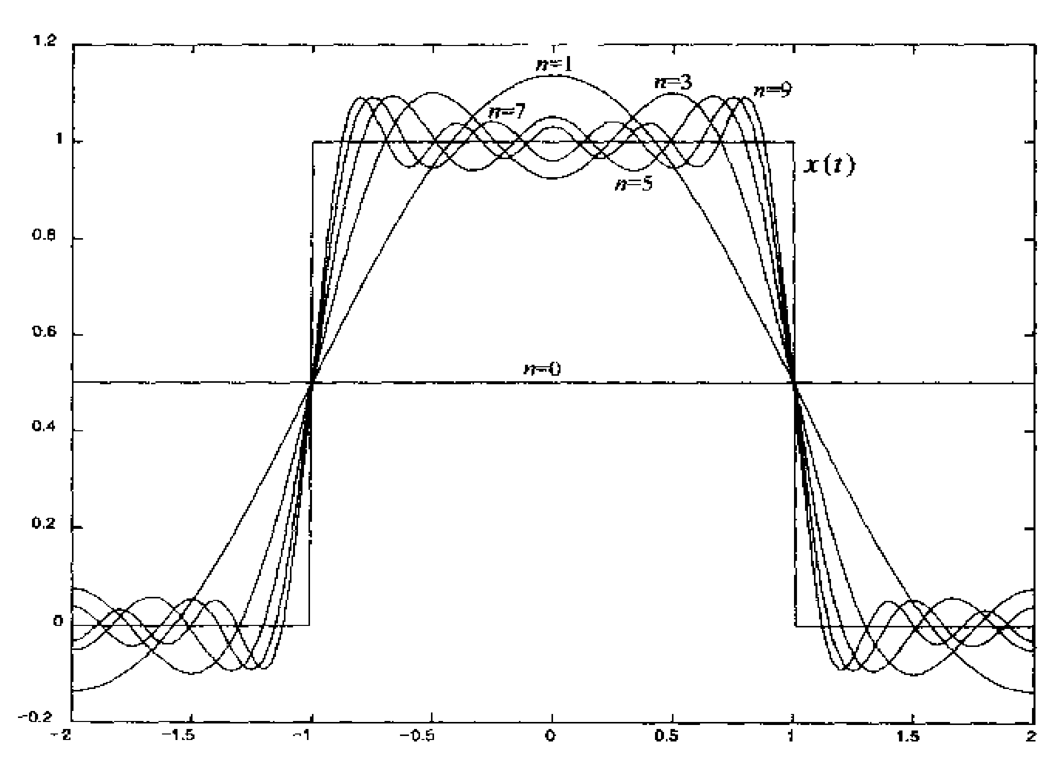


图 1.3　在解说题 1.1 中，对矩形脉冲的各种傅里叶级数的近似

下面是画出信号离散谱的 MATLAB 脚本。

M 文件

```
% MATLAB script for Illustrative Problem 1.1.
n=[-20:1:20];
x=abs(sinc(n/2));
stem(n,x);
```

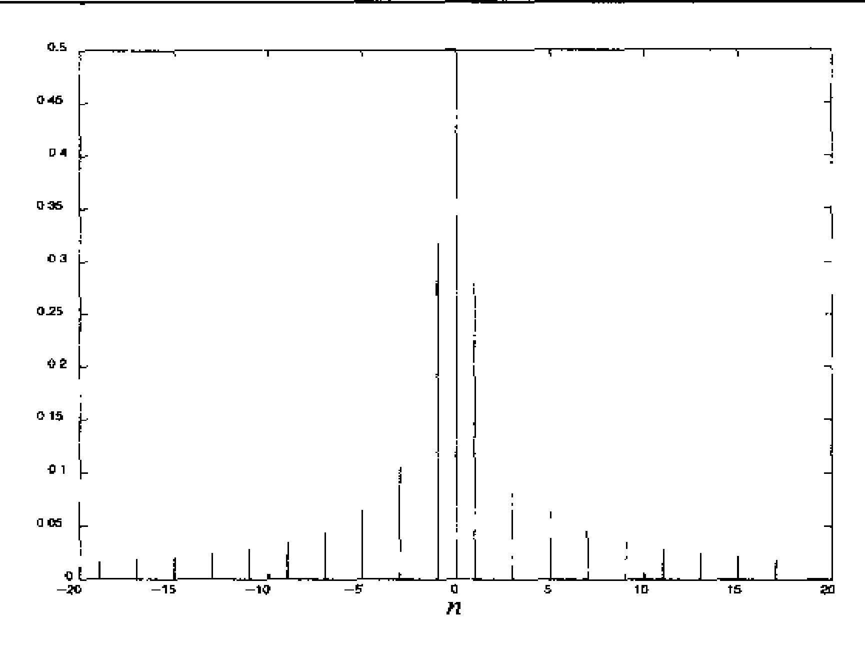


图 1.4 解说题 1.1 中的信号的离散谱

当信号 $x(t)$在一个周期内按图 1.5 的方式由 a, b 之间给出时,在这个区间[a, b]内的信号由一个 m 文件给出,其傅里叶级数系数可用下面的 m 文件 fseries.m 求得。

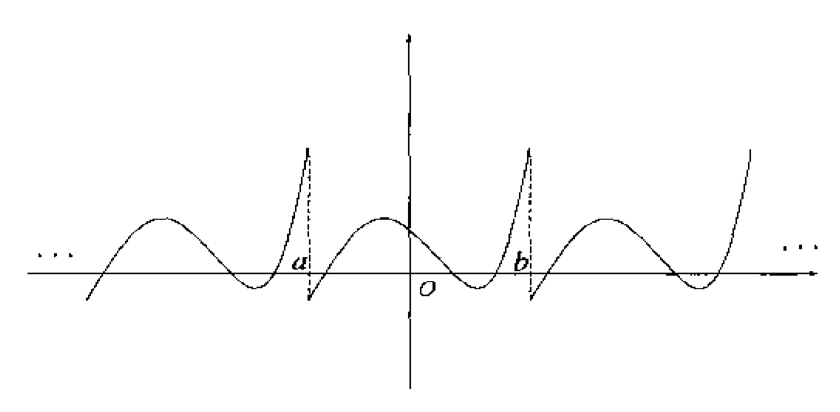


图 1.5 一种周期信号

m 文件

```
function xx=fseries(funfcn,a,b,n,tol,p1,p2,p3)
%FSERIES         Returns the Fourier series coefficients.
%                XX=FSERIES(FUNFCN,A,B,N,TOL,P1,P2,P3)
%                funfcn=the given function, in an m-file.
%                It can depend on up to three parameters
%                p1, p2, and p3. The function is given
%                over one period extending from 'a' to 'b'
%                xx=vector of length n+1 of Fourier series
%                coefficients, xx0,xx1,...,xxn.
%                p1,p2,p3=parameters of funfcn.
%                tol=the error level.

j=sqrt(-1);
args0=[ ];
for nn=1:nargin-5
  args0=[args0,',p',int2str(nn)];
```

```
end
args=[args0,')'];
t=b-a;
xx(1)=eval(['1/(',num2str(t),').*quad(funfcn,a,b,tol,[]',args]);

for i=1:n
  new_fun = 'exp_fnct';
  args=[',',num2str(i),',',num2str(t),args0,')'];
  xx(i+1)=eval(['1/(',num2str(t),').*quad(new_fun,a,b,tol,[],funfcn',args]);
end
```

解说题

解说题 1.2　[幅度谱和相位谱]

求出并画出周期为 8，$x(t)=\Lambda(t)$，$|t|\leqslant 4$ 的周期信号的离散幅度谱和相位谱。

解　题

因为信号已由 m 文件 lambda.m 给出，可以通过选定区间 $[a, b]=[-4, 4]$ 求出各系数。应该注意，由 m 文件 fseries.m 求得的傅里叶系数是对应于 n 的非负值的，但由于现在 $x(t)$ 是实值的，从而有 $x_{-n}=x_n^*$。图 1.6 给出了 $n=24$ 时该信号的幅度谱和相位谱。

求出并画出幅度谱和相位谱的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 1.2.
echo on
fnct='lambda';
a=-4;
b=4;
n=24;
tol=0.1;
xx=fseries(fnct,a,b,n,tol);
xx1=xx(n+1:-1:2);
xx1=[conj(xx1),xx];
absxx1=abs(xx1);
pause % Press any key to see a plot of the magnitude spectrum.
n1=[-n:n];
stem(n1,absxx1)
title('The Discrete Magnitude Spectrum')
phasexx1=angle(xx1);
pause % Press any key to see a plot of the phase spectrum.
stem(n1,phasexx1)
title('The Discrete Phase Spectrum')
```

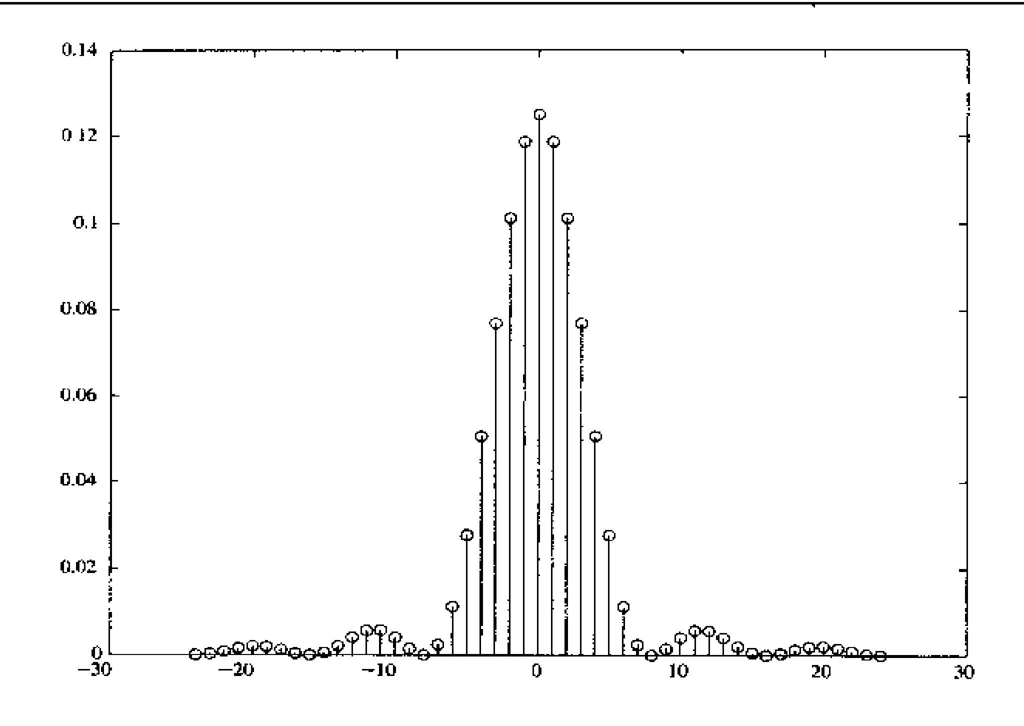
解说题

解说题 1.3　[幅度谱和相位谱]

求出并画出周期为 12，在区间 $[-6, 6]$ 内，$x(t)$ 为：

$$x(t)=\frac{1}{\sqrt{2\pi}}e^{-t^2/2}$$

的周期信号的幅度谱和相位谱。$x(t)$ 如图 1.7 所示。



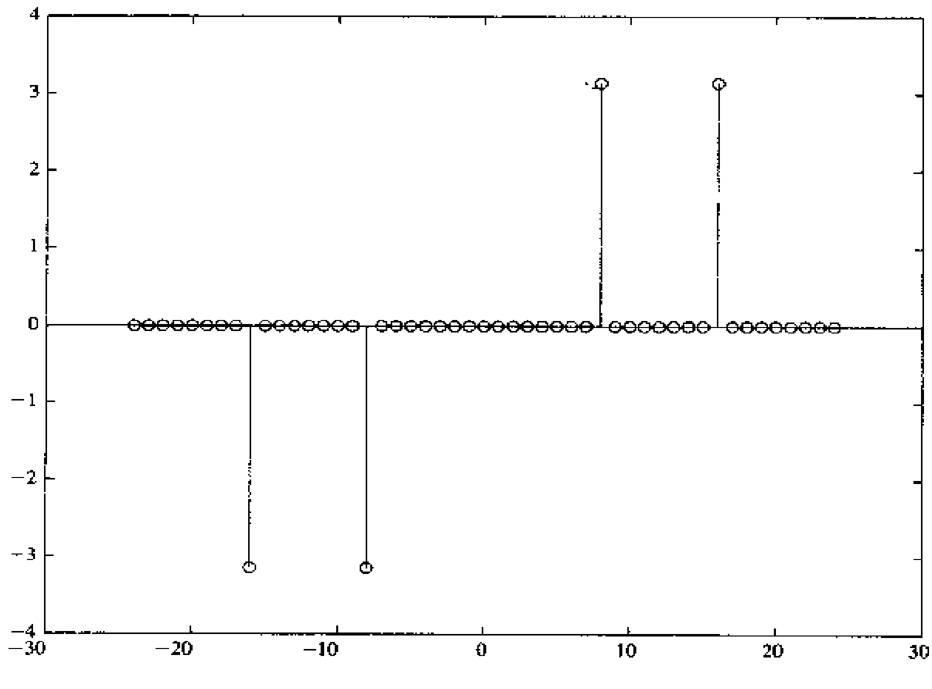


图 1.6　解说题 1.2 中信号的幅度谱和相位谱

题　解

该信号就是一个零均值、单位方差高斯(正态)随机变量的密度函数,这个函数由 m 文件 normal.m 给出。该文件要求有两个参数 m 和 s,即该随机变量的均值和标准偏差;在本题中分别为 0 和 1。因此,可以用下列 MATLAB 脚本求得如图 1.8 所示的幅度谱和相位谱。

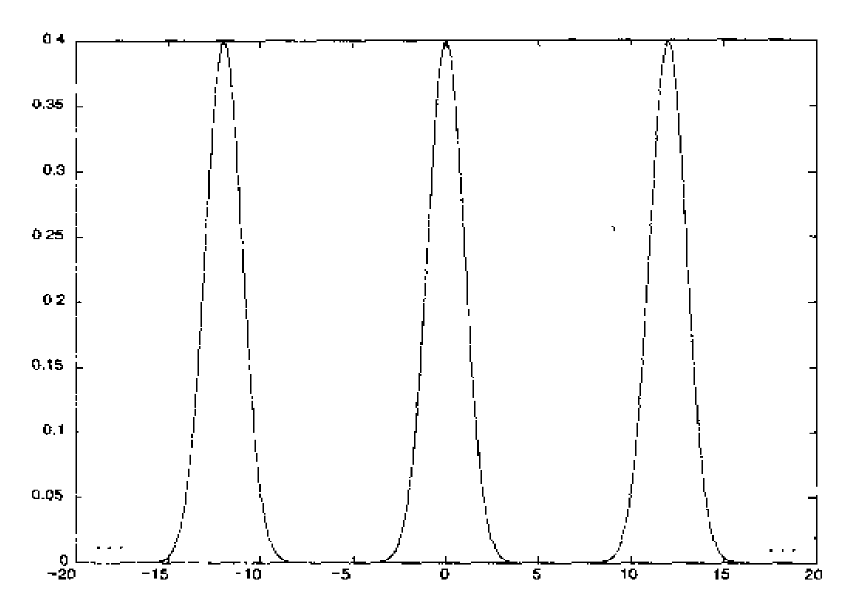


图 1.7　解说题 1.3 中的周期信号

m 文件

```
% MATLAB script for Illustrative Problem 1.3.
echo on
fnct='normal';
a=-6;
b=6;
n=24;
tol=0.1;
xx=fseries(fnct,a,b,n,tol,0,1);
xx1=xx(n+1:-1:2);
xx1=[conj(xx1),xx];
absxx1=abs(xx1);
pause % Press any key to see a plot of the magnitude.
n1=[-n:n];
stem(n1,absxx1)
title('The Discrete Magnitude Spectrum')
phasexx1=angle(xx1);
pause % Press any key to see a plot of the phase.
stem(n1,phasexx1)
title('The Discrete Phase Spectrum')
```

1.2.1　周期信号和 LTI 系统

当一个周期信号 $x(t)$通过某一 LTI 系统时，如图 1.9 所示，输出 $y(t)$也是周期的，并且通常与输入信号具有相同的周期[①]（为什么?），因此也有一个傅里叶级数展开式。

如果 $x(t)$和 $y(t)$展开为：

$$x(t) = \sum_{n=-\infty}^{\infty} x_n e^{j2\pi nt/T_0} \tag{1.2.26}$$

① 这里说的是"通常"与输入信号具有相同的周期。你能举出一个输出的周期与输入的周期不同的例子吗?

图 1.8 解说题 1.3 中信号的幅度谱和相位谱

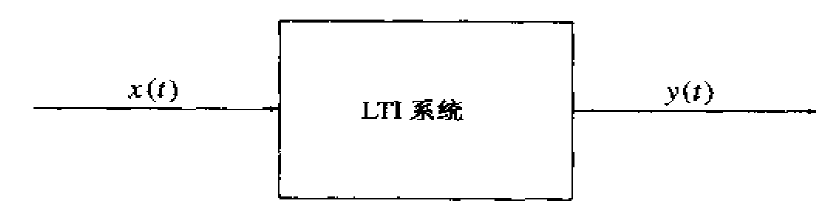


图 1.9 周期信号通过 LTI 系统

$$y(t) = \sum_{n=-\infty}^{\infty} y_n e^{j2\pi nt/T_0} \tag{1.2.27}$$

那么 $x(t)$和 $y(t)$的傅里叶级数系数之间的关系就能用卷积积分求得为：

$$\begin{aligned} y(t) &= \int_{-\infty}^{\infty} x(t-\tau)h(\tau)d\tau \\ &= \int_{-\infty}^{\infty} \sum_{n=-\infty}^{\infty} x_n e^{j2\pi n(t-\tau)/T_0} h(\tau)d\tau \\ &= \sum_{n=-\infty}^{\infty} x_n \left(\int_{-\infty}^{\infty} h(\tau)e^{-j2\pi n\tau/T_0}d\tau\right) e^{j2\pi nt/T_0} \\ &= \sum_{n=-\infty}^{\infty} y_n e^{j2\pi nt/T_0} \end{aligned} \tag{1.2.28}$$

由上述关系，有：

$$y_n = x_n H\left(\frac{n}{T_0}\right) \tag{1.2.29}$$

其中 $H(f)$记为该 LTI 系统的传递函数①，它是系统冲激响应 $h(t)$的傅里叶变换：

$$H(f) = \int_{-\infty}^{\infty} h(t)e^{-j2\pi ft}dt \tag{1.2.30}$$

解说题

解说题 1.4　[周期信号的过滤]

有一个三角脉冲串 $x(t)$，周期 $T_0=2$，在一个周期内 $x(t)$定义为：

$$\Lambda(t) = \begin{cases} t+1, & -1 \leqslant t \leqslant 0 \\ -t+1, & 0 < t \leqslant 1 \\ 0, & \text{其余 } t \end{cases} \tag{1.2.31}$$

1. 求 $x(t)$的傅里叶级数系数。
2. 画出 $x(t)$的离散谱。
3. 设想这个信号通过某 LTI 系统，该系统的冲激响应是：

$$h(t) = \begin{cases} t, & 0 \leqslant t < 1 \\ 0, & \text{其余 } t \end{cases} \tag{1.2.32}$$

画出输出 $y(t)$的离散谱。$x(t)$和 $h(t)$的图如图 1.10 所示。

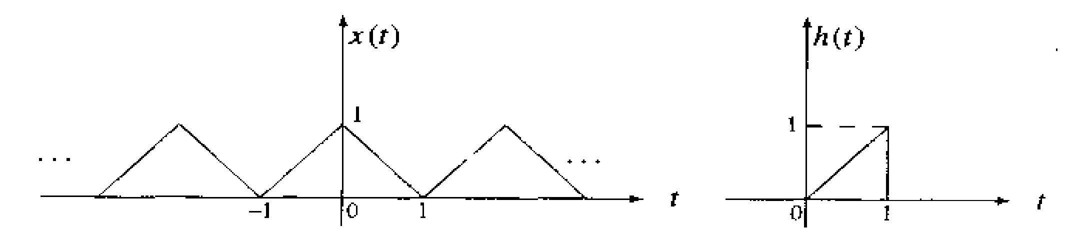


图 1.10　输入信号和系统冲激响应

① 也称为系统的频率响应。

题 解

1. 根据已给出的 $x(t)$,有:

$$x_n = \frac{1}{T_0}\int_{-T_0/2}^{T_0/2} x(t)e^{-j2\pi nt/T_0}\,dt \tag{1.2.33}$$

$$= \frac{1}{2}\int_{-1}^{1} \Lambda(t)e^{-j\pi nt}\,dt \tag{1.2.34}$$

$$= \frac{1}{2}\int_{-\infty}^{\infty} \Lambda(t)e^{-j\pi nt}\,dt \tag{1.2.35}$$

$$= \frac{1}{2}\mathscr{F}[\Lambda(t)]_{f=n/2} \tag{1.2.36}$$

$$= \frac{1}{2}\mathrm{sinc}^2\left(\frac{n}{2}\right) \tag{1.2.37}$$

其中已经用到 $\Lambda(t)$在区间$[-1,1]$外为零和 $\Lambda(t)$的傅里叶变换是 $\mathrm{sinc}^2(f)$。利用 $\Lambda(t)$的表示式进行分部积分,也能得出这一结果。显然,对于除 $n=0$ 以外全部为偶数的 n,其中 $x_n=0$。

2. $x(t)$的离散谱如图 1.11 所示。

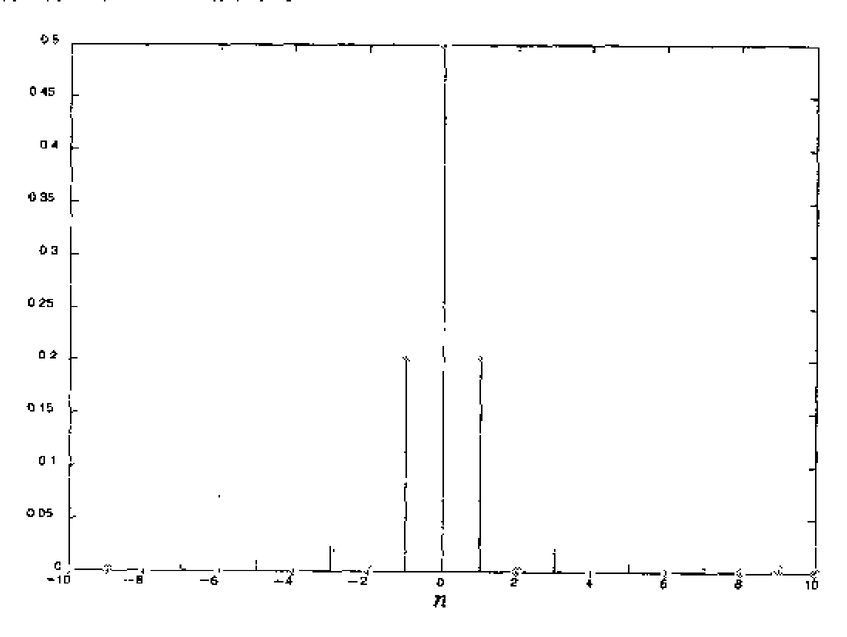


图 1.11 信号的离散谱

3. 首先必须导出系统的传递函数 $H(f)$。虽然这可以用解析方法来完成,但我们还是愿意采用数值方法。所得到的传递函数幅度以及 $H(n/T_0)=H(n/2)$的幅度如图 1.12 所示。为得到输出的离散谱,使用如下关系:

$$y_n = x_n H\left(\frac{n}{T_0}\right) \tag{1.2.38}$$

$$= \frac{1}{2}\mathrm{sinc}^2\left(\frac{n}{2}\right)H\left(\frac{n}{2}\right) \tag{1.2.39}$$

图 1.13 即为输出的离散谱。

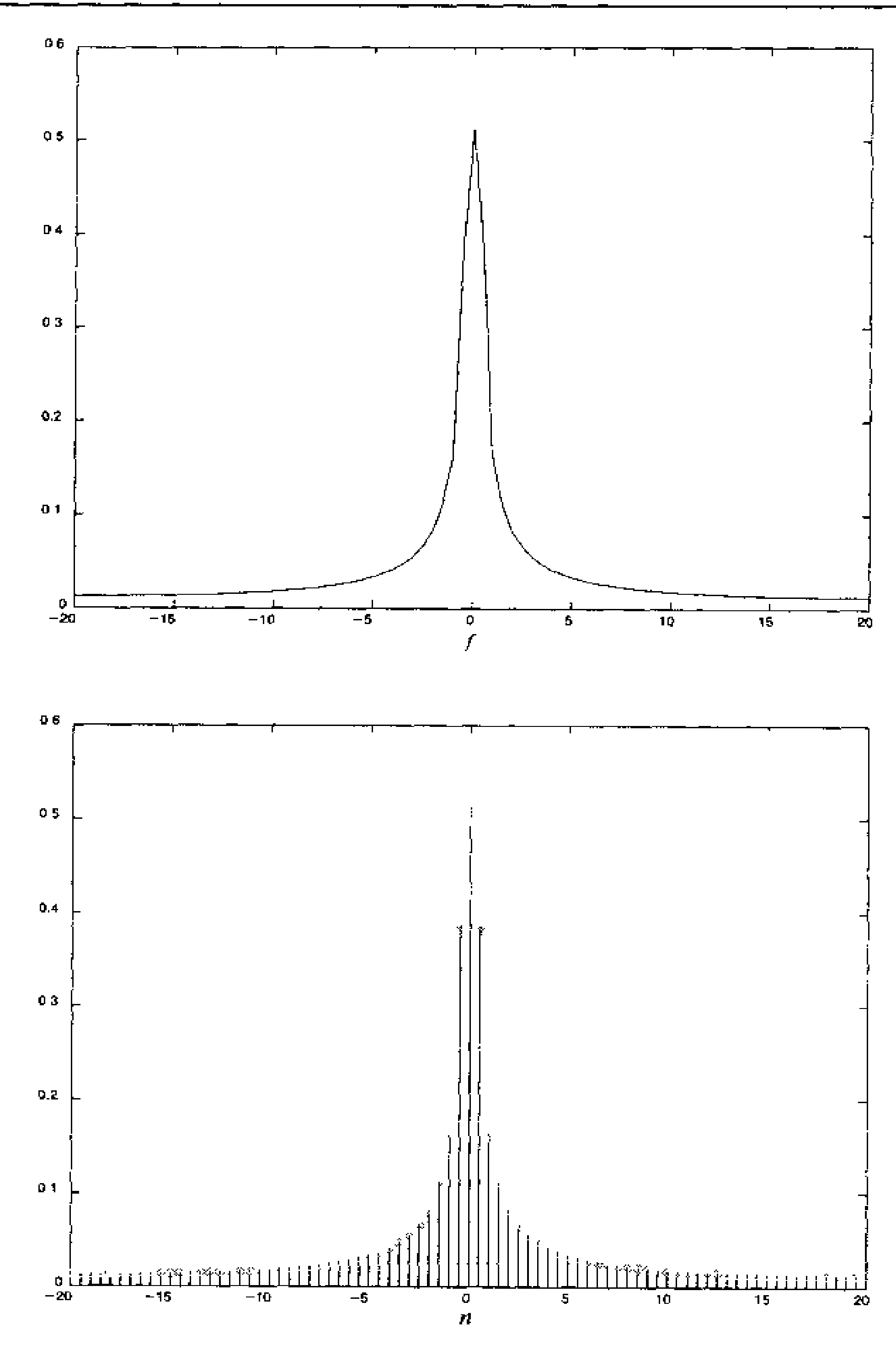


图 1.12　该 LTI 系统的传递函数和 $H(n/2)$ 的幅度

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 1.4.
echo on
n=[-20:1:20];
% Fourier series coefficients of x(t) vector
```

```
x=.5*(sinc(n/2)).^2;
% sampling interval
ts=1/40;
% time vector
t=[-.5:ts:1.5];
% impulse response
fs=1/ts;
h=[zeros(1,20),t(21:61),zeros(1,20)];
% transfer function
H=fft(h)/fs;
% frequency resolution
df=fs/80;
f=[0:df:fs]-fs/2;
% rearrange H
H1=fftshift(H);
y=x.*H1(21:61);
% Plotting commands follow.
```

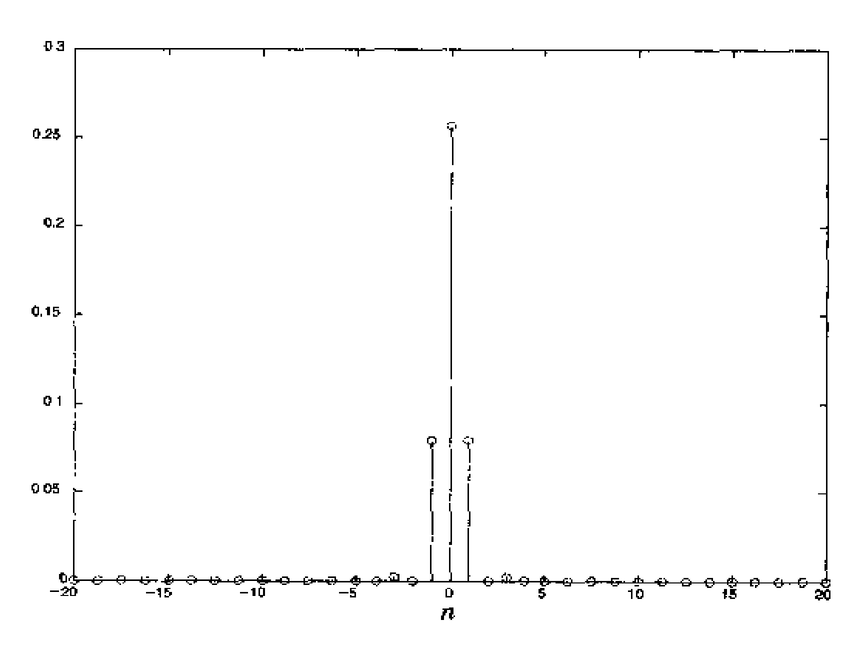


图 1.13 输出的离散谱

1.3 傅里叶变换

傅里叶变换是傅里叶级数对非周期信号的推广。一个满足狄里赫利条件[1]的信号 $x(t)$，其傅里叶变换(记为 $X(f)$,或$\mathcal{F}[x(t)]$)定义为：

$$\mathcal{F}[x(t)] = X(f) = \int_{-\infty}^{\infty} x(t)\mathrm{e}^{-j2\pi ft}\,\mathrm{d}t \tag{1.3.1}$$

$X(f)$的傅里叶逆变换是 $x(t)$,为：

$$\mathcal{F}^{-1}[X(f)] = x(t) = \int_{-\infty}^{\infty} X(f)\mathrm{e}^{j2\pi ft}\,\mathrm{d}f \tag{1.3.2}$$

如果 $x(t)$为实信号,那么 $X(f)$就满足埃尔米特对称,即

$$X(-f)=X^*(f) \tag{1.3.3}$$

傅里叶变换满足几个性质，其中最重要的性质综合如下：

1. **线性**。两个或更多个信号线性组合的傅里叶变换就是相应单个傅里叶变换的线性组合，即

$$\mathcal{F}[\alpha x_1(t)+\beta x_2(t)]=\alpha\mathcal{F}[x_1(t)]+\beta\mathcal{F}[x_2(t)] \tag{1.3.4}$$

2. **对偶性**。若 $X(f)=\mathcal{F}[x(t)]$，则

$$\mathcal{F}[X(t)]=x(-f) \tag{1.3.5}$$

3. **时移**。在时域内的移位会导致频域内的相移。若 $X(f)=\mathcal{F}[x(t)]$，则

$$\mathcal{F}[x(t-t_0)]=\mathrm{e}^{-\mathrm{j}2\pi f t_0}X(f) \tag{1.3.6}$$

4. **尺度变换**。在时域内的扩展会导致频域内的压缩，反之亦然。若 $X(f)=\mathcal{F}[x(t)]$，则

$$\mathcal{F}[x(at)]=\frac{1}{|a|}X\left(\frac{f}{a}\right),\qquad a\neq 0 \tag{1.3.7}$$

5. **调制**。在时域内乘以指数相应于在频域内的频移。若 $X(f)=\mathcal{F}[x(t)]$，则

$$\begin{cases}\mathcal{F}[\mathrm{e}^{\mathrm{j}2\pi f_0 t}x(t)]=X(f-f_0)\\ \mathcal{F}[x(t)\cos(2\pi f_0 t)]=\dfrac{1}{2}[X(f-f_0)+X(f+f_0)]\end{cases} \tag{1.3.8}$$

6. **微分**。在时域内的微分相应于在频域内乘以 $\mathrm{j}2\pi f$，若 $X(f)=\mathcal{F}[x(t)]$，则

$$\mathcal{F}[x'(t)]=\mathrm{j}2\pi f X(f) \tag{1.3.9}$$

$$\mathcal{F}\left[\frac{\mathrm{d}^n}{\mathrm{d}t^n}x(t)\right]=(\mathrm{j}2\pi f)^n X(f) \tag{1.3.10}$$

7. **卷积**。时域卷积等效于频域相乘，反之亦然。若 $X(f)=\mathcal{F}[x(t)]$ 和 $Y(f)=\mathcal{F}[y(t)]$，则

$$\mathcal{F}[x(t)*y(t)]=X(f)Y(f) \tag{1.3.11}$$

$$\mathcal{F}[x(t)y(t)]=X(f)*Y(f) \tag{1.3.12}$$

8. **帕塞瓦尔（Parseval）定理**。若 $X(f)=\mathcal{F}[x(t)]$ 和 $Y(f)=\mathcal{F}[y(t)]$，则

$$\int_{-\infty}^{\infty}x(t)y^*(t)\mathrm{d}t=\int_{-\infty}^{\infty}X(f)Y^*(f)\mathrm{d}f \tag{1.3.13}$$

$$\int_{-\infty}^{\infty}|x(t)|^2\mathrm{d}t=\int_{-\infty}^{\infty}|X(f)|^2\mathrm{d}f \tag{1.3.14}$$

第二个关系也称**瑞利（Rayleigh）定理**。

表 1.1 给出了若干最有用的傅里叶变换对。表中 $u_{-1}(t)$代表单位阶跃函数，$\delta(t)$代表冲激信号，$\mathrm{sgn}(t)$是**符号函数**，定义为：

$$\mathrm{sgn}(t)=\begin{cases}1, & t>0\\ 0, & t=0\\ -1, & t<0\end{cases} \tag{1.3.15}$$

而 $\delta^{(n)}(t)$代表冲激信号的 n 阶导数。

表 1.1 傅里叶变换对表

$x(t)$	$X(f)$
$\delta(t)$	1
1	$\delta(f)$
$\delta(t-t_0)$	$e^{-j2\pi f t_0}$
$e^{j2\pi f_0 t}$[1]	$\delta(f-f_0)$
$\cos(2\pi f_0 t)$	$\frac{1}{2}\delta(f-f_0)+\frac{1}{2}\delta(f+f_0)$
$\sin(2\pi f_0 t)$	$\frac{1}{2j}\delta(f-f_0)-\frac{1}{2j}\delta(f+f_0)$
$\Pi(t)$	$\text{sinc}(f)$
$\text{sinc}(t)$	$\Pi(f)$
$\Lambda(t)$	$\text{sinc}^2(f)$
$\text{sinc}^2(t)$	$\Lambda(f)$
$e^{-\alpha t}u_{-1}(t),\alpha>0$	$\frac{1}{\alpha+j2\pi f}$
$te^{-\alpha t}u_{-1}(t),\alpha>0$	$\frac{1}{(\alpha+j2\pi f)^2}$
$e^{-\alpha\|t\|},\alpha>0$	$\frac{2\alpha}{\alpha^2+(2\pi f)^2}$
$e^{-\pi t^2}$	$e^{-\pi f^2}$
$\text{sgn}(t)$	$\frac{1}{j\pi f}$
$u_{-1}(t)$	$\frac{1}{2}\delta(f)+\frac{1}{j2\pi f}$
$\delta'(t)$	$j2\pi f$
$\delta^{(n)}(t)$	$(j2\pi f)^n$
$\sum_{n=-\infty}^{\infty}\delta(t-nT_0)$	$\frac{1}{T_0}\sum_{n=-\infty}^{\infty}\delta\left(f-\frac{n}{T_0}\right)$

对于一个周期信号 $x(t)$,周期为 T_0,其傅里叶级数系数由 x_n 给出,即

$$x(t)=\sum_{n=-\infty}^{\infty}x_n e^{j2\pi nt/T_0}$$

$x(t)$的傅里叶变换为:

$$\begin{aligned}X(f)&=\mathcal{F}[x(t)]\\&=\mathcal{F}\left[\sum_{n=-\infty}^{\infty}x_n e^{j2\pi nt/T_0}\right]\\&=\sum_{n=-\infty}^{\infty}x_n\,\mathcal{F}\left[e^{j2\pi nt/T_0}\right]\\&=\sum_{n=-\infty}^{\infty}x_n\delta\left(f-\frac{n}{T_0}\right)\end{aligned}\tag{1.3.16}$$

换句话说,一个周期信号的傅里叶变换由发生在原信号基波频率整倍数(谐波)频率处的冲激组成。

利用**截断信号** $x_{T_0}(t)$的傅里叶变换 $X_{T_0}(f)$,也能表示傅里叶级数,系数 x_n 为:

[1] 原著此处为"$e^{j2\pi f t_0}$",似有误。——译者注。

$$x_n = \frac{1}{T_0} X_{T_0}\left(\frac{n}{T_0}\right) \tag{1.3.17}$$

这里 $x_{T_0}(t)$定义为：

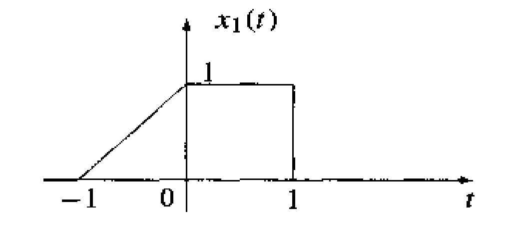
$$x_{T_0}(t) = \begin{cases} x(t), & -\frac{T_0}{2} < t \leq \frac{T_0}{2} \\ 0, & 其余\ t \end{cases} \tag{1.3.18}$$

信号的傅里叶变换称为信号的**频谱**。信号的频谱一般是个复函数 $X(f)$，因此为了画出频谱通常要用两个图：幅度谱 $|X(f)|$ 和相位谱 $\angle X(f)$。

解说题

解说题 1.5　[傅里叶变换]

画出图 1.14 中信号 $x_1(t)$和 $x_2(t)$的幅度谱和相位谱。



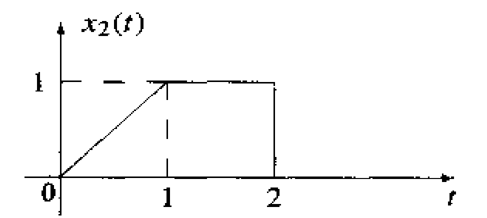


图 1.14　信号 $x_1(t)$和 $x_2(t)$

题　解

因为这两个信号除了一个时移外是相同的，所以会有相同的幅度谱。相同的幅度谱和两个相位谱分别画在同一个坐标轴上，如图 1.15 和图 1.16 所示。

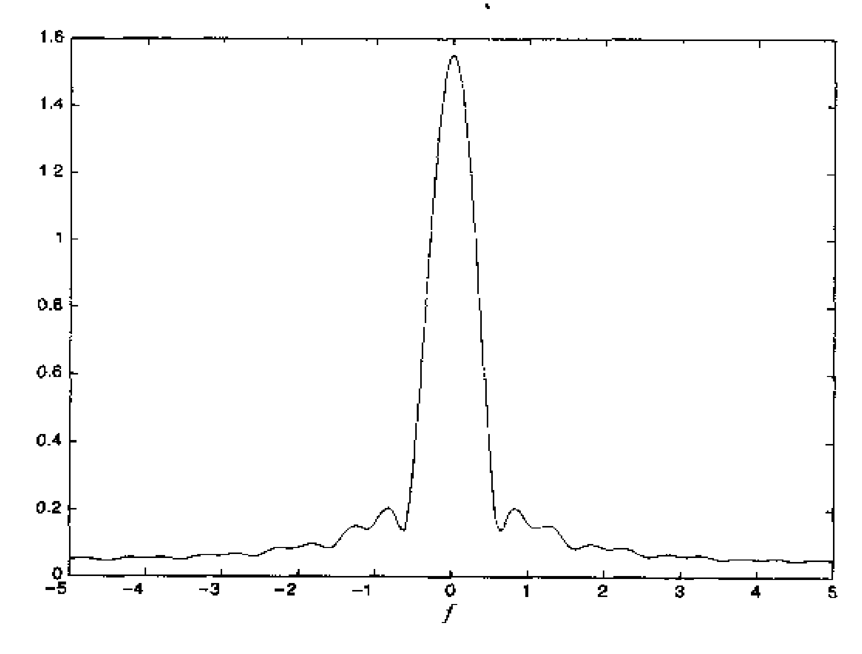


图 1.15　信号 $x_1(t)$和 $x_2(t)$的幅度谱

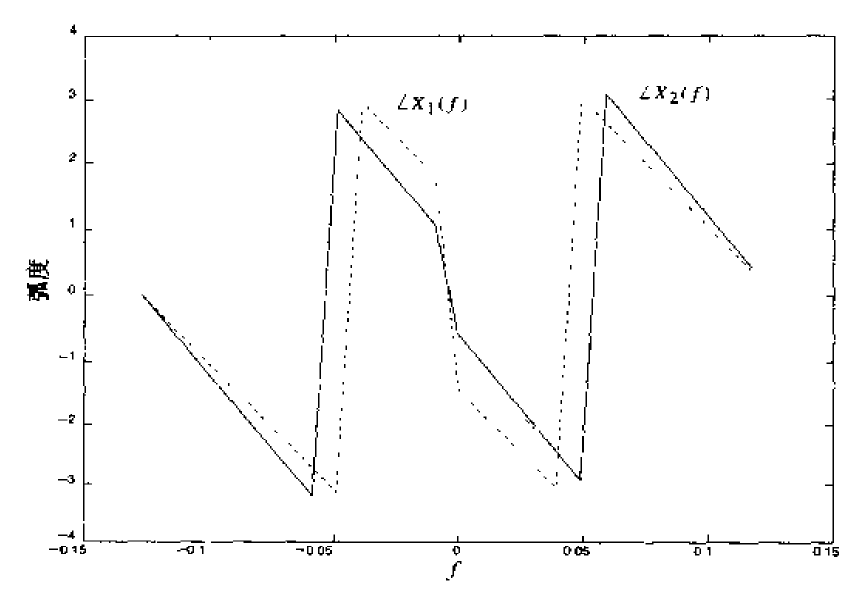


图 1.16 信号 $x_1(t)$和 $x_2(t)$的相位谱

这个题的 MATLAB 脚本如下所示。在 1.3.1 节中将说明如何用 MATLAB 求得一个信号的傅里叶变换。

m 文件

```
% MATLAB script for Illustrative Problem 5.1.
df=0.01;
fs=10;
ts=1/fs;
t=[-5:ts:5];
x1=zeros(size(t));
x1(41:51)=t(41:51)+1;
x1(52:61)=ones(size(x1(52:61)));
x2=zeros(size(t));
x2(51:71)=x1(41:61);
[X1,x11,df1]=fftseq(x1,ts,df);
[X2,x21,df2]=fftseq(x2,ts,df);
X11=X1/fs;
X21=X2/fs;
f=[0:df1:df1*(length(x11)-1)]-fs/2;
plot(f,fftshift(abs(X11)))
figure
plot(f(500:525),fftshift(angle(X11(500:525))),f(500:525),fftshift(angle(X21(500:525))),'--')
```

1.3.1 采样定理

在信号与系统分析中,采样定理是最重要的结果之一,它构成了连续时间信号和离散时间信号之间的关系的基础。采样定理指的是,一个带限信号,即信号的傅里叶变换,对某个 W 有 $|f| > W$为零,可以完全用在间隔为 T_s 的时刻所取得的样本值来表示,只要 $T_s \leqslant 1/(2W)$。如果采样在间隔为 $T_s = 1/(2W)$(称为**奈奎斯特间隔**或**奈奎斯特率**)下完成,那么 $x(t)$就能按

$$x(t) = \sum_{n=-\infty}^{\infty} x(nT_s)\text{sinc}(2W(t - nT_s)) \tag{1.3.19}$$

由样本值 $\{x[n]=x(nT_s)\}_{n=-\infty}^{\infty}$ 重建。这一结果基于已采样波形 $x_\delta(t)$ 定义为：

$$x_\delta(t)=\sum_{n=-\infty}^{\infty}x(nT_s)\delta(t-nT_s) \tag{1.3.20}$$

它的傅里叶变换为：

$$X_\delta(f)=\begin{cases}\dfrac{1}{T_s}\displaystyle\sum_{n=-\infty}^{\infty}X\left(f-\dfrac{n}{T_s}\right), & \text{对全部 } f\\ \dfrac{1}{T_s}X(f), & |f|<W\end{cases} \tag{1.3.21}$$

所以将它通过一个带宽为 W，通带内增益为 T_s 的低通滤波器就可以将原信号恢复。

图 1.17 表示的是式(1.3.19)在 $T_s=1$ 和 $\{x[n]\}_{n=-3}^{3}=\{1,1,-1,2,-2,1,2\}$ 时的结果，即

$$x(t)=\text{sinc}(t+3)+\text{sinc}(t+2)-\text{sinc}(t+1)+2\text{sinc}(t)$$
$$-2\text{sinc}(t-1)+\text{sinc}(t-2)+2\text{sinc}(t-2)$$

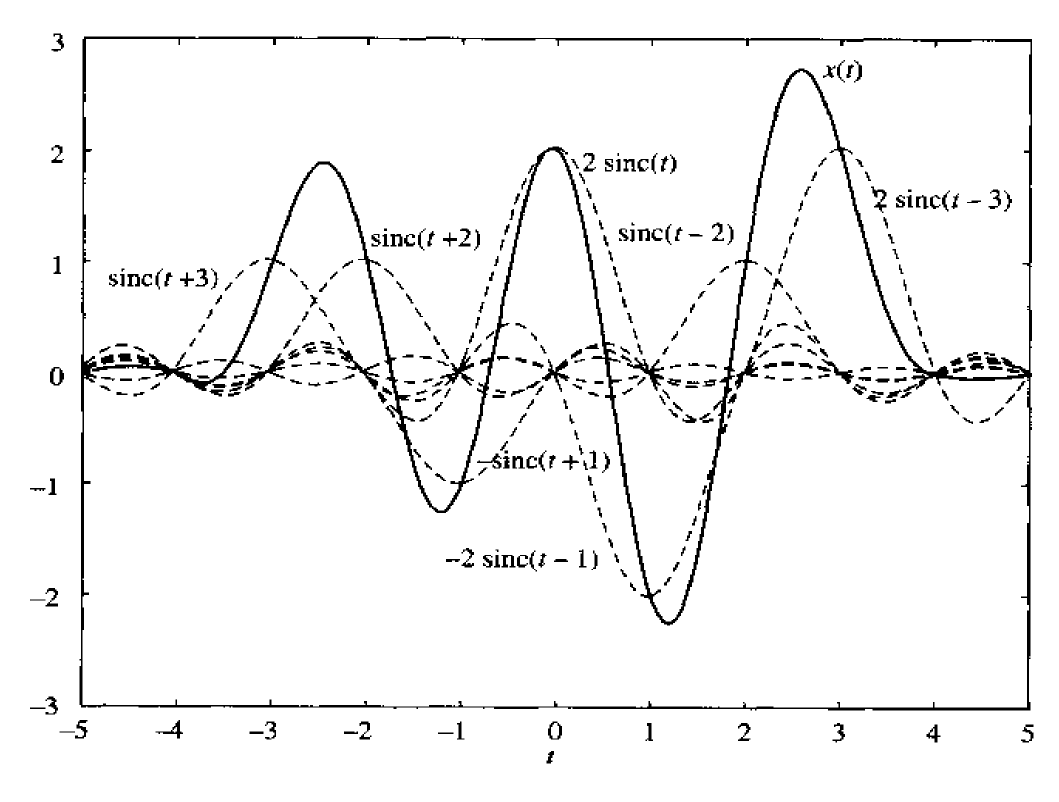


图 1.17　采样定理的表示

离散时间序列 $x[n]$ 的**离散傅里叶变换(DFT)**表示为：

$$X_d(f)=\sum_{n=-\infty}^{\infty}x[n]e^{-j2\pi fnT_s} \tag{1.3.22}$$

将它与式(1.3.21)进行比较后可得：

$$X(f)=T_sX_d(f),\qquad |f|<W \tag{1.3.23}$$

它给出了模拟信号的傅里叶变换和它的对应已采样信号的离散傅里叶变换之间的关系。

经由著名的**快速傅里叶变换(FFT)**算法，可以对离散傅里叶变换完成数值计算。在这个算法中，在 T_s 间隔点上所取得的 $x(t)$ 的 N 个样本序列用做该信号的表示，其结果是在频率间

隔$[0,f_s]$上的$X_d(f)$的N个样本序列，其中$f_s=1/T_s=2W$是奈奎斯特频率。当这些样本以$\Delta f=f_s/N$分隔开时，Δf值就给出了所得傅里叶变换的频率分辨率。如果输入序列的长度N是2的幂，FFT算法在计算上是高效的。在很多情况下，如果序列的长度不是2的幂，可以通过补零等技术使其成为2的幂。值得注意的是，因为FFT算法实质上只给出了已采样信号的DFT，为了得到模拟信号的傅里叶变换，还必须使用式(1.3.23)。这意味着在计算出FFT之后还必须将其乘以T_s，或者等效为除以f_s，以得出原模拟信号的傅里叶变换。

下面给出的MATLAB函数fftseq.m将时间序列m，采样间隔t_s和要求的频率分辨率df作为输入，就得到了长度为2的幂的序列、这个序列M的FFT以及所要的频率分辨率。

m文件

```
function [M,m,df]=fftseq(m,ts,df)
%            [M,m,df]=fftseq(m,ts,df)
%            [M,m,df]=fftseq(m,ts)
%FFTSEQ      generates M, the FFT of the sequence m.
%            The sequence is zero-padded to meet the required frequency resolution df.
%            ts is the sampling interval. The output df is the final frequency resolution.
%            Output m is the zero-padded version of input m. M is the FFT.
fs=1/ts;
if nargin == 2
  n1=0;
else
  n1=fs/df;
end
n2=length(m);
n=2^(max(nextpow2(n1),nextpow2(n2)));
M=fft(m,n);
m=[m,zeros(1,n-n2)];
df=fs/n;
```

解说题

解说题 1.6 ［傅里叶变换的解析求解和数值求解］

信号$x(t)$为：

$$x(t)=\begin{cases} t+2, & -2\leqslant t\leqslant -1 \\ 1, & -1<t\leqslant 1 \\ -t+2, & 1<t\leqslant 2 \\ 0, & \text{其余 } t \end{cases} \tag{1.3.24}$$

如图1.18所示。

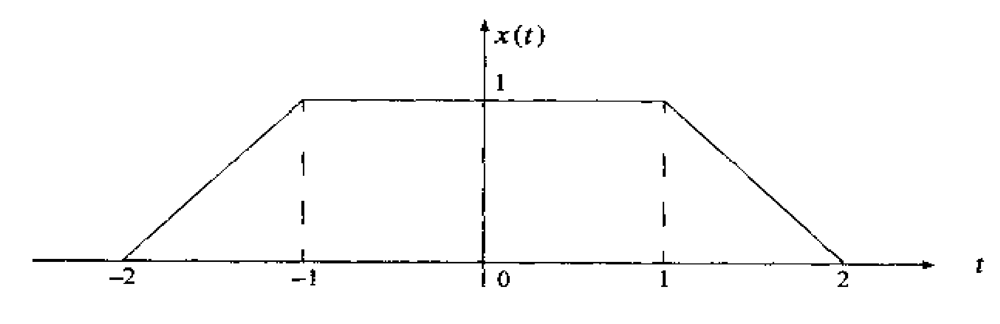


图1.18 信号$x(t)$

1. 用解析法求 $x(t)$的傅里叶变换,并画出 $x(t)$的频谱。

2. 用 MATLAB 求出该傅里叶变换的数值解,并画出结果。

题　解

1. 信号 $x(t)$可以写成:

$$x(t) = 2\Lambda\left(\frac{t}{2}\right) - \Lambda(t) \tag{1.3.25}$$

因此有:

$$X(f) = 4\text{sinc}^2(2f) - \text{sinc}^2(f) \tag{1.3.26}$$

其中已经用了线性变换、尺度变换和 $\Lambda(t)$的傅里叶变换是 $\text{sinc}^2(f)$这些条件。显然,该傅里叶变换是实函数。幅度谱如图 1.19 所示。

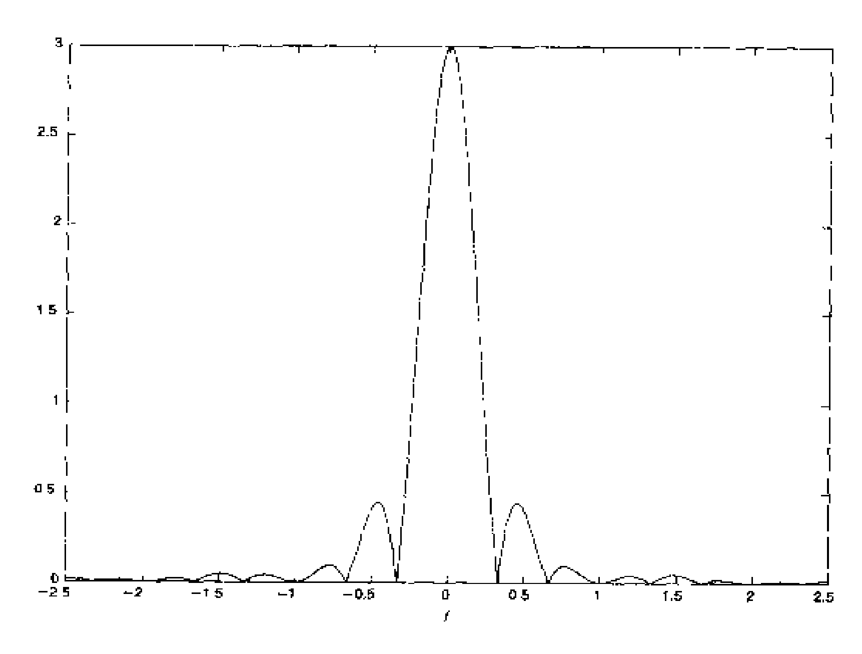


图 1.19　用解析法求得的 $x(t)$幅度谱

2. 为了用 MATLAB 求傅里叶变换,首先要给出该信号带宽的大致估计。因为这个信号相对比较平滑,它的带宽正比于信号持续时间的倒数,该信号的持续时间是 4。为了安全可靠起见,带宽取为信号持续时间的倒数的 10 倍,即

$$\text{BW} = 10 \times \frac{1}{4} = 2.5 \tag{1.3.27}$$

奈奎斯特频率是带宽的 2 倍,即 5,采样间隔 $T_s = 1/f_s = 0.2$。现考虑在区间 $[-4, 4]$内,以 T_s对信号采样。有了这些选取之后,就可以使用 fftseq.m 函数的简单 MATLAB 脚本得出数值解 FFT。已经选定的所要求的频率分辨率是 0.01 Hz。而用 fftseq.m 得到的频率分辨率是 0.0098 Hz,它满足本题的要求。长度为 41 的信号向量 x 通过补零到 256 长度,以满足频率分辨率的要求,同时进行高效计算使长度成为 2 的幂。傅里叶变换的幅度谱如图 1.20 所示。

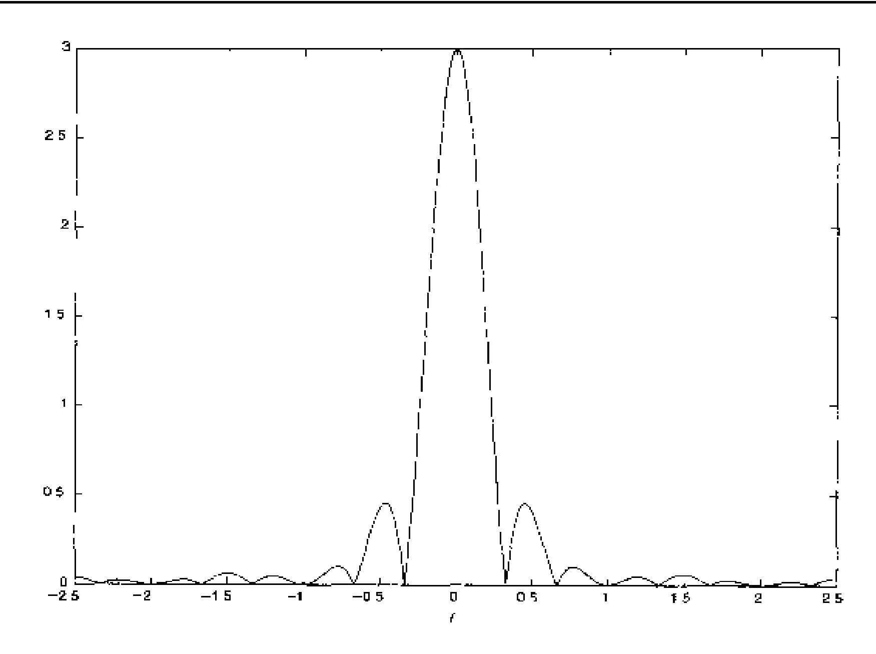


图 1.20 用数值法求得的 $x(t)$幅度谱

这个题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 1.6.
echo on
ts=0.2;                                         % set parameters
fs=1/ts;
df=0.01;
x=[zeros(1,10),[0:0.2:1],ones(1,9),[1:-0.2:0],zeros(1,10)];
[X,x,df1]=fftseq(x,ts,df);                      % Derive the FFT
X1=X/fs;                                        % Scaling
f=[0:df1:df1*(length(x)-1)]-fs/2;              % frequency vector for FFT
f1=[-2.5:0.001:2.5];                            % frequency vector for analytic approach
y=4*(sinc(2*f1)).^2-(sinc(f1)).^2;             % exact Fourier transform
pause % Press a key to see the plot of the Fourier transform derived analytically.
clf
subplot(2,1,1)
plot(f1,abs(y));
xlabel('Frequency')
title('Magnitude-spectrum of x(t) derived analytically')
pause % Press a key to see the plot of the Fourier transform derived numerically.
subplot(2,1,2)
plot(f,fftshift(abs(X1)));
xlabel('Frequency')
title('Magnitude spectrum of x(t) derived numerically')
```

1.3.2 LTI 系统的频域分析

当输入信号为 $x(t)$时，冲激响应为 $h(t)$的 LTI 系统的输出由卷积积分给出：

$$y(t) = x(t) * h(t) \tag{1.3.28}$$

应用卷积定理可得：

$$Y(f) = X(f)H(f) \tag{1.3.29}$$

其中

$$H(f)=\mathscr{F}[h(t)]=\int_{-\infty}^{\infty}h(t)\mathrm{e}^{-\mathrm{j}2\pi ft}\mathrm{d}t \tag{1.3.30}$$

是该系统的传递函数。式(1.3.29)可写成如下形式：

$$\begin{cases}|Y(f)|=|X(f)||H(f)|\\ \angle Y(f)=\angle X(f)+\angle H(f)\end{cases} \tag{1.3.31}$$

该式给出了输入和输出幅度谱和相位谱之间的关系。

解说题

解说题 1.7　[LTI 系统在频域内的分析]

信号 $x(t)$如图 1.21 所示，它由若干直线段和一个正弦的一部分组成。

1. 求信号的 FFT，并画出它。
2. 若该信号通过带宽为 1.5 Hz 的理想低通滤波器，求该滤波器的输出，并画出它。
3. 若该信号通过一个冲激响应为：

$$h(t)=\begin{cases}t, & 0\leqslant t<1\\ 1, & 1\leqslant t\leqslant 2\\ 0, & 其余\ t\end{cases} \tag{1.3.32}$$

的滤波器，画出该滤波器的输出。

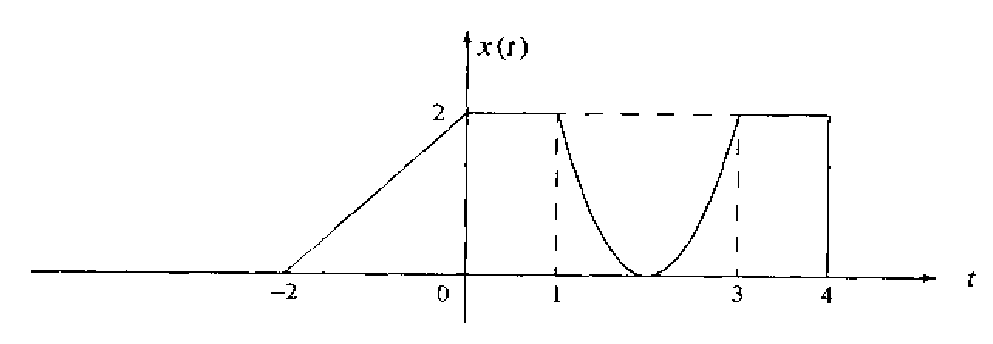


图 1.21　信号 $x(t)$

题　解

首先要导出信号正弦部分的表达式。这个正弦的一半周期是 2，所以频率为 $f_0=\frac{1}{4}=0.25$ Hz。信号的幅度为 2，而且被向上提升了 2。它的一般表达式为 $2\cos(2\pi\times0.25t+\theta)+2=2\cos(0.5\pi t+\theta)+2$。相位 θ 值可用如下边界条件来确定：

$$2+2\cos(0.5\pi t+\theta)|_{t=2}=0 \tag{1.3.33}$$

即 $\theta=0$。因此，这个信号可写成：

$$x(t)=\begin{cases}t+2 & -2\leqslant t\leqslant 0\\ 2^{①}, & 0<t\leqslant 1\\ 2+2\cos(0.5\pi t), & 1<t\leqslant 3\\ 1, & 3<t\leqslant 4\\ 0, & 其余\ t\end{cases} \tag{1.3.34}$$

① 原著此处为 1，但根据图 1.21 应为 2，原著似有误。——译者注

有了这个信号的完全表达式,就能继续做下去。

1. 信号带宽已经选为 5 Hz,要求的频率分辨率是 0.01 Hz。图 1.22 显示的是信号幅度谱的图。

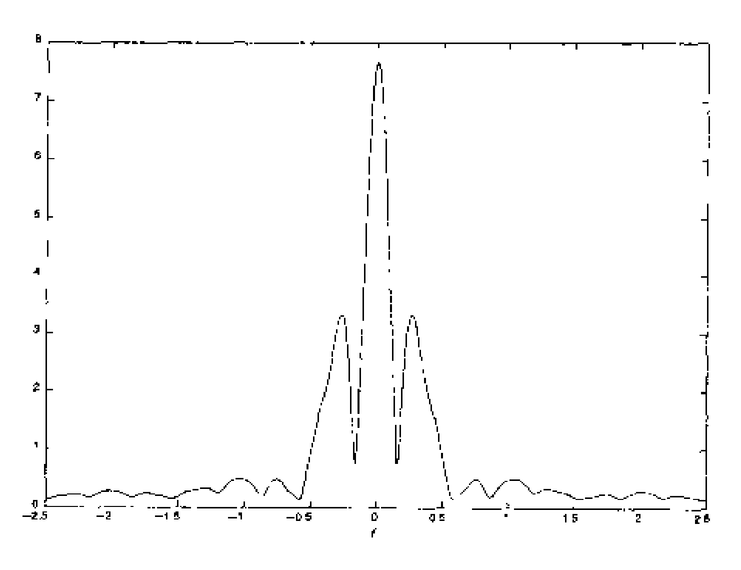


图 1.22 信号的幅度谱

2. 现在 $f_s = 5$ Hz。因为低通滤波器的带宽是 1.5 Hz,它的传递函数为:

$$H(f)=\begin{cases}1, & 0\leqslant f\leqslant 1.5\\ 0, & 1.5<f\leqslant 3.5\\ 1, & 3.5<f\leqslant 5\end{cases} \tag{1.3.35}$$

将它乘以 $X(f)$ 就得出 $Y(f)$,即输出的傅里叶变换。用这个传递函数所得的输出如图 1.23 所示。

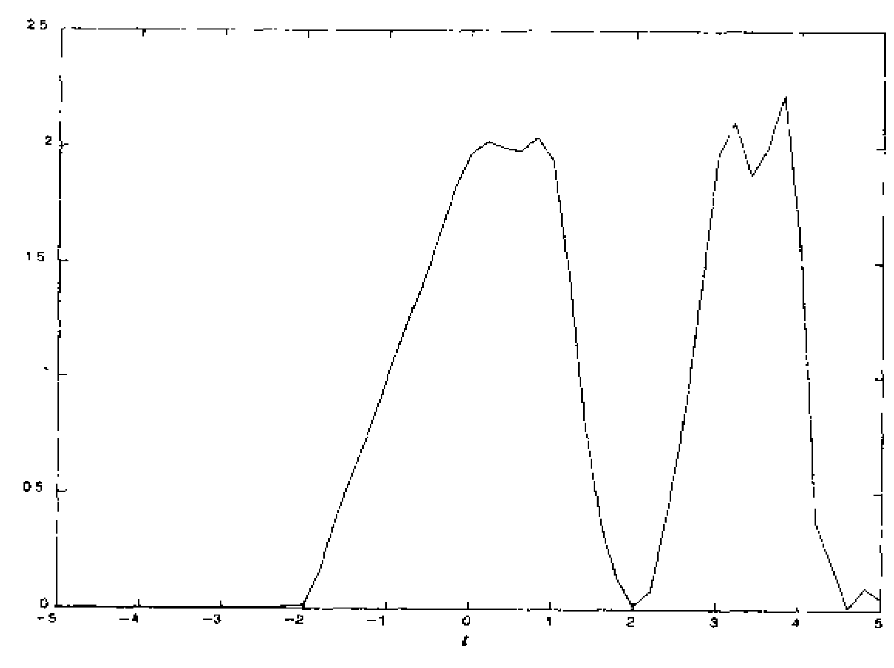


图 1.23 低通滤波器的输出

3. 现在,用简单卷积所得的滤波器输出如图 1.24 所示。

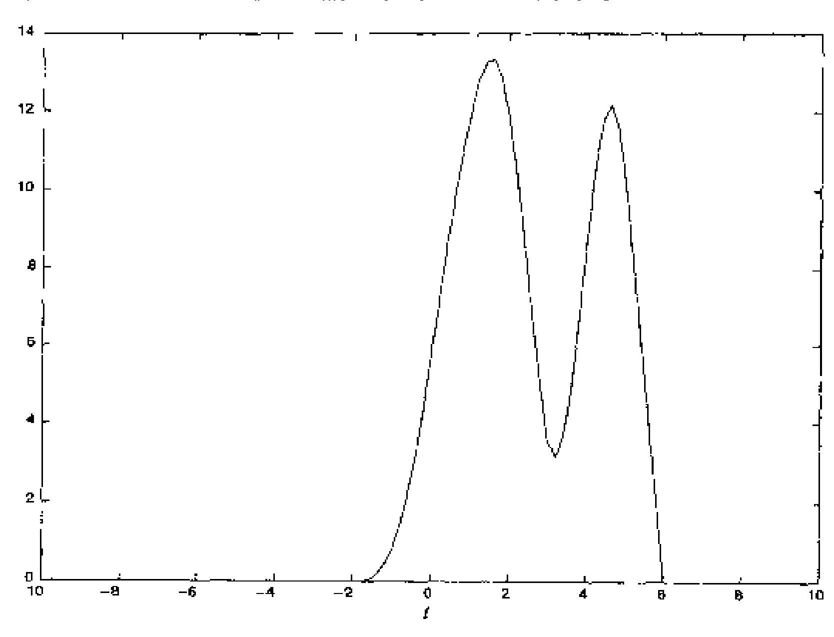


图 1.24 解说题 1.7 中第三部分的输出信号

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 1.7.
echo on
df=0.01;                                  % freq. resolution
fs=5;                                     % sampling frequency
ts=1/fs;                                  % sampling interval
t=[-5:ts:5];                              % time vector
x=zeros(1,length(t));                     % input signal initiation
x(16:26)=t(16:26)+2;
x(27:31)=2*ones(1,5);
x(32:41)=2+2*cos(0.5*pi*t(32:41));
x(42:46)=2*ones(1,5);
% Part 1
[X,x1,df1]=fftseq(x,ts,df);               % spectrum of the input
f=[0:df1:df1*(length(x1)-1)]-fs/2;        % frequency vector
X1=X/fs;                                  % scaling
% Part 2
% filter transfer function
H=[ones(1,ceil(1.5/df1)),zeros(1,length(X)-2*ceil(1.5/df1)),ones(1,ceil(1.5/df1))];
Y=X.*H;                                   % output spectrum
y1=ifft(Y);                               % output of the filter
% Part 3
% LTI system impulse response
h=[zeros(1,ceil(5/ts)),t(ceil(5/ts)+1:ceil(6/ts)),ones(1,ceil(7/ts)-ceil(6/ts)),zeros(1,51-ceil(7/ts))];
y2=conv(h,x);                             % output of the LTI system
pause   % Press a key to see spectrum of the input.
plot(f,fftshift(abs(X1)))
pause   % Press a key to see the output of the lowpass filter.
plot(t,abs(y1(1:length(t))));
pause   % Press a key to see the output of the LTI system.
plot([-10:ts:10],y2);
```

1.4 功率和能量

一个实信号 $x(t)$的能量和功率分别记为 E_X 和 P_X,定义为:

$$\begin{cases} E_X = \int_{-\infty}^{\infty} x^2(t)\mathrm{d}t \\ P_X = \lim_{T\to\infty} \frac{1}{T}\int_{-T/2}^{T/2} x^2(t)\mathrm{d}t \end{cases} \tag{1.4.1}$$

具有有限能量的信号称为**能量信号**,而具有正的和有限功率的信号称为**功率信号**[①]。例如,$x(t)=\Pi(t)$是能量信号的一个例子,而 $x(t)=\cos(t)$则是功率信号的一个例子。所有周期信号[②] 都是功率信号。一个能量信号的**能谱密度**给出了该信号在各个频率上的能量分布,它由

$$\mathcal{G}_X(f) = |X(f)|^2 \tag{1.4.2}$$

给出。因此有:

$$E_X = \int_{-\infty}^{\infty} \mathcal{G}_X(f)\mathrm{d}f \tag{1.4.3}$$

利用卷积定理,有:

$$\mathcal{G}_X(f) = \mathcal{F}[R_X(\tau)] \tag{1.4.4}$$

式 $R_X(\tau)$是 $x(t)$的**自相关函数**,对于实值信号定义为:

$$\begin{aligned} R_X(\tau) &= \int_{-\infty}^{\infty} x(t)x(t+\tau)\mathrm{d}t \\ &= x(\tau) * x(-\tau) \end{aligned} \tag{1.4.5}$$

对于功率信号,定义**时间平均自相关函数**为:

$$R_X(\tau) = \lim_{T\to\infty} \frac{1}{T}\int_{-T/2}^{T/2} x(t)x(t+\tau)\mathrm{d}t \tag{1.4.6}$$

定义**功率谱密度**为:

$$\mathcal{S}_X(f) = \mathcal{F}[R_X(\tau)] \tag{1.4.7}$$

总功率是功率谱密度的积分:

$$P_X = \int_{-\infty}^{\infty} \mathcal{S}_X(f)\mathrm{d}f \tag{1.4.8}$$

对于周期为 T_0 的周期信号 $x(t)$的特例,其傅里叶级数系数为 x_n,功率谱密度为:

$$\mathcal{S}_X(f) = \sum_{n=-\infty}^{\infty} |x_n|^2 \delta\left(f - \frac{n}{T_0}\right) \tag{1.4.9}$$

这意味着全部功率都集中在基波频率的各次谐波上,在第 n 次谐波(n/T_0)的功率是$|x_n|^2$,即相应傅里叶级数系数的模平方。

---

① 有些信号既非能量信号又非功率信号,如 $x(t)=e^t u_{-1}(t)$。

② 惟一的例外是处处为零的信号。

当信号 $x(t)$ 通过传递函数为 $H(f)$ 的滤波器时，输出的能量谱密度或功率谱密度由下式求得：

$$\begin{cases}\mathscr{G}_Y(f) = |H(f)|^2 \mathscr{G}_X(f) \\ \mathcal{S}_Y(f) = |H(f)|^2 \mathcal{S}_X(f)\end{cases} \tag{1.4.10}$$

若使用离散时间(已采样)信号，与式(1.4.1)相当的能量和功率关系变成：

$$\begin{cases}E_X = T_s \sum_{n=-\infty}^{\infty} x^2[n] \\ P_X = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} x^2[n]\end{cases} \tag{1.4.11}$$

如果使用 FFT，即序列的长度是有限长的并且是重复的，则有：

$$\begin{cases}E_X = T_s \sum_{n=0}^{N-1} x^2[n] \\ P_X = \frac{1}{N} \sum_{n=0}^{N-1} x^2[n]\end{cases} \tag{1.4.12}$$

以下的 MATLAB 函数 power.m 给出了信号向量的功率。

m 文件

```
function p=spower(x)
%              p=spower(x)
%SPOWER        returns the power in signal x
p=(norm(x)^2)/length(x);
```

如果 $X_d(f)$ 是序列 $x[n]$ 的 DFT，那么 $x(t)$ 的能量谱密度(等效的模拟信号)可以用式(1.3.23)求得，并给出：

$$\mathscr{G}_X(f) = T_s^2 |X_d(f)|^2 \tag{1.4.13}$$

其中 T_s 是采样间隔。序列 $x[n]$ 的功率谱密度最容易用 MATLAB 函数 spectrum.m 求得。

解说题

解说题 1.8 [功率和功率谱]

有一个持续期为 10，且为两个单位振幅的正弦信号之和的信号 $x(t)$，两个信号中的一个频率为47 Hz，另一个为 219 Hz：

$$x(t) = \begin{cases}\cos(2\pi\times 47t) + \cos(2\pi\times 219t), & 0\leqslant t\leqslant 10 \\ 0, & \text{其余 } t\end{cases}$$

该信号以采样率 1000 样本/s 进行采样。用 MATLAB 求该信号的功率和功率谱密度。

题 解

用 MATLAB 函数 spower.m，该信号的功率求得为 1.0003 W。用 spectrum.m 和 specplot.m 可以画出该信号的功率谱密度，如图 1.25 所示。图中功率谱的双峰对应于信号中存在的两个频率。

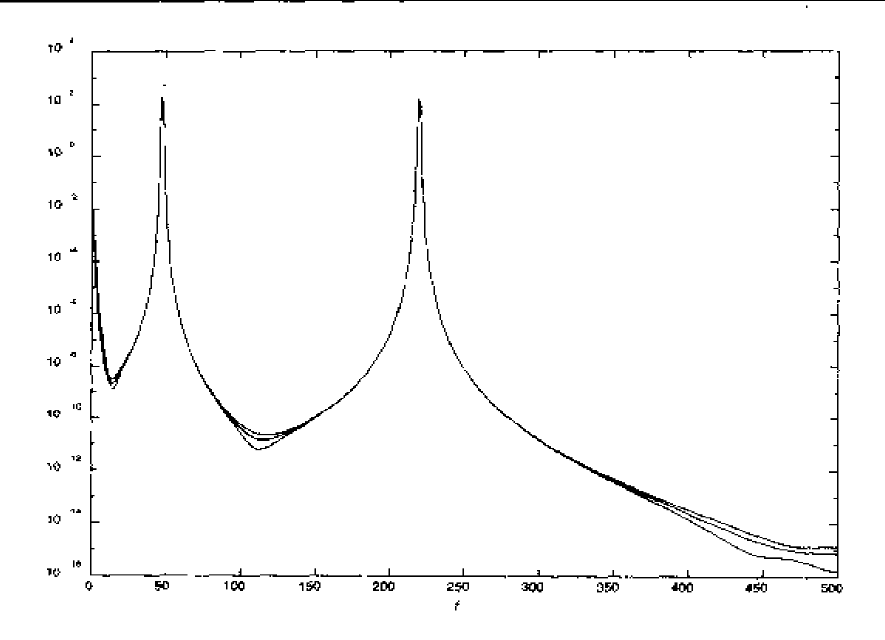


图 1.25 由频率为 $f_1 = 47$ Hz 和 $f_2 = 219$ Hz 的两个正弦信号组成的信号的功率谱密度

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 1.8.
ts=0.001;
fs=1/ts;
t=[0:ts:10];
x=cos(2*pi*47*t)+cos(2*pi*219*t);
p=spower(x);
psd=spectrum(x,1024);
pause   % Press a key to see the power in the signal.
p
pause   % Press a key to see the power spectrum.
specplot(psd,fs)
```

1.5 带通信号的低通等效

一个带通信号就是其全部频率分量都位于某个**中心频率** f_0(自然也在 $-f_0$)附近的信号。换句话说,带通信号有 $x(f) \equiv 0$, $|f \pm f_0| > W$,这里 $W \ll f_0$。一个低通信号是其频率分量均位于零频率附近的信号,即对于 $|f| > W$,有 $X(f) \equiv 0$。

对应于一个带通信号 $x(t)$,可以定义**解析信号** $z(t)$,它的傅里叶变换由下式给出:

$$Z(f) = 2u_{-1}(f)X(f) \tag{1.5.1}$$

其中 $u_{-1}(f)$是单位阶跃函数。这一关系在时域中可以写为:

$$z(t) = x(t) + \mathrm{j}\hat{x}(t) \tag{1.5.2}$$

其中 $\hat{x}(t)$记为 $x(t)$的**希尔伯特变换**,即 $\hat{x}(t) = x(t) * (1/\pi t)$,在频域中则可以写为:

$$\hat{X}(f) = -\mathrm{j}\,\mathrm{sgn}(f)X(f) \tag{1.5.3}$$

在 MATLAB 中，希尔伯特变换函数记为 hilbert.m，它产生复序列 $z(t)$。$z(t)$的实部是原序列，而它的虚部则是原序列的希尔伯特变换。

信号 $x(t)$的低通等效记为 $x_l(t)$，用 $z(t)$表示为：

$$x_l(t) = z(t)\mathrm{e}^{-\mathrm{j}2\pi f_0 t} \tag{1.5.4}$$

根据这个关系，有：

$$\begin{cases} x(t) = \mathrm{Re}[x_l(t)\mathrm{e}^{\mathrm{j}2\pi f_0 t}] \\ \hat{x}(t) = \mathrm{Im}[x_l(t)\mathrm{e}^{\mathrm{j}2\pi f_0 t}] \end{cases} \tag{1.5.5}$$

在频域中则为：

$$X_l(f) = Z(f+f_0) = 2u_{-1}(f+f_0)X(f+f_0) \tag{1.5.6}$$

和

$$X_l(f) = X(f-f_0) + X^*(-f-f_0) \tag{1.5.7}$$

一般来说，一个实带通信号的低通等效是一个复信号；它的实部 $x_c(t)$称为 $x(t)$的同相分量，而它的虚部 $x_s(t)$称为 $x(t)$的正交分量，即

$$x_l(t) = x_c(t) + \mathrm{j}x_s(t) \tag{1.5.8}$$

利用同相和正交分量，有：

$$\begin{cases} x(t) = x_c(t)\cos(2\pi f_0 t) - x_s(t)\sin(2\pi f_0 t) \\ \hat{x}(t) = x_s(t)\cos(2\pi f_0 t) + x_c(t)\sin(2\pi f_0 t) \end{cases} \tag{1.5.9}$$

如果将 $x_l(t)$用极坐标表示，则有：

$$x_l(t) = V(t)\mathrm{e}^{\mathrm{j}\Theta(t)} \tag{1.5.10}$$

其中 $V(t)$和 $\Theta(t)$称为信号 $x(t)$的**包络**和**相位**。利用这两个量，有

$$x(t) = V(t)\cos(2\pi f_0 t + \Theta(t)) \tag{1.5.11}$$

包络和相位可以表示为：

$$\begin{cases} V(t) = \sqrt{x_c^2(t) x_s^2(t)} \\ \Theta(t) = \arctan \dfrac{x_s(t)}{x_c(t)} \end{cases} \tag{1.5.12}$$

或者等效为：

$$\begin{cases} V(t) = \sqrt{x^2(t) + \hat{x}^2(t)} \\ \Theta(t) = \arctan \dfrac{\hat{x}(t)}{x(t)} - 2\pi f_0 t \end{cases} \tag{1.5.13}$$

由上面的关系显然可以知道，包络与 f_0 的选取无关，而相位则与这个选取有关。

我们已经写了一些简单的 MATLAB 文件，用于产生解析信号，产生一个信号的低通表示，产生同相和正交分量以及包络和相位。这些 MATLAB 函数分别是 analytic.m，loweq.m，quadcomp.m 以及 env_phas.m。这些函数如下所示。

m文件

```
function z=analytic(x)
%             z=analytic(x)
%ANALYTIC     returns the analytic signal corresponding to signal x
%
z=hilbert(x);
```

m文件

```
function xl=loweq(x,ts,f0)
%             xl=loweq(x,ts,f0)
%LOWEQ        returns the lowpass equivalent of the signal x
%             f0 is the center frequency.
%             ts is the sampling interval.
%
t=[0:ts:ts*(length(x)-1)];
z=hilbert(x);
xl=z.*exp(-j*2*pi*f0*t);
```

m文件

```
function [xc,xs]=quadcomp(x,ts,f0)
%             [xc,xs]=quadcomp(x,ts,f0)
%QUADCOMP     Returns the in-phase and quadrature components of
%             the signal x. f0 is the center frequency. ts is the
%             sampling interval.
%
z=loweq(x,ts,f0);
xc=real(z);
xs=imag(z);
```

m文件

```
function [v,phi]=env_phas(x,ts,f0)
%             [v,phi]=env_phas(x,ts,f0)
%                  v=env_phas(x,ts,f0)
%ENV_PHAS     returns the envelope and the phase of the bandpass signal x
%             f0 is the center frequency.
%             ts is the sampling interval.
%
if nargout == 2
  z=loweq(x,ts,f0);
  phi=angle(z);
end
v=abs(hilbert(x));
```

解说题

解说题 1.9 [带通到低通变换]

信号 $x(t)$ 为:

$$x(t)=\text{sinc}(100t)\cos(2\pi\times200t) \tag{1.5.14}$$

1. 画出该信号和它的幅度谱。

2. 用 $f_0=200$ Hz 求 $x(t)$的低通等效，并画出它的幅度谱。

3. 假设 $f_0=100$ Hz，求 $x(t)$的低通等效，并画出它的幅度谱。

题 解

选取采样间隔 $t_s=0.001$ s，采样频率则为 $f_s=1/t_s=1000$ Hz。选取所期望的频率分辨率 $\mathrm{d}f=0.5$ Hz，可有如下结果：

1. 信号图形和它的幅度谱如图 1.26 所示。这两幅图都是用 MATLAB 产生的。

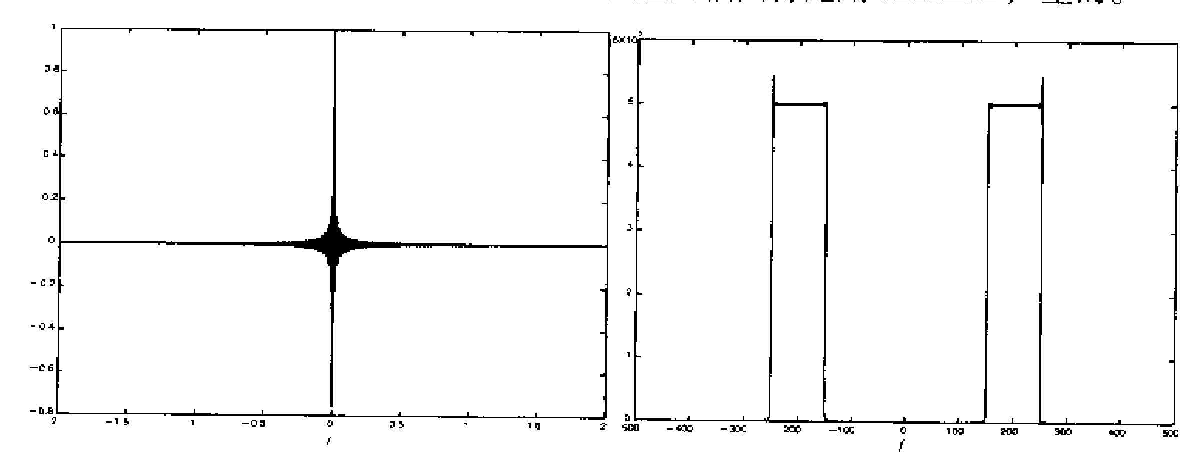


图 1.26 信号 $x(t)$和它的幅度谱

2. 选取 $f_0=200$ Hz，利用 loweq.m 函数可求得 $x(t)$的低通等效。然后用 fftseq.m 得到它的频谱，图 1.27 给出了它的幅度谱。

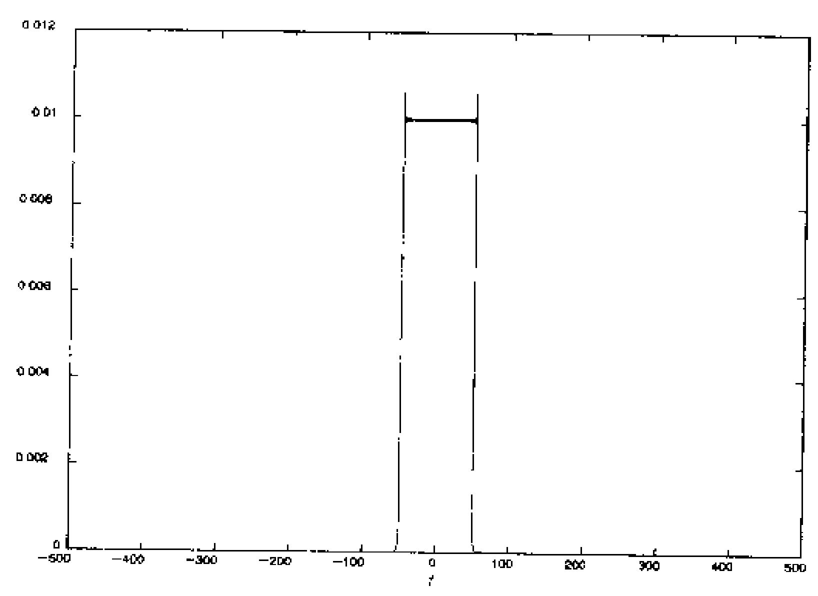


图 1.27 当 $f_0=200$ Hz 时，解说题 1.9 中 $x(t)$的低通等效的幅度谱

由该图可见，幅度谱在这种情况下是偶函数，因为有：

$$x(t)=\mathrm{Re}[\mathrm{sinc}(100t)\mathrm{e}^{\mathrm{j}2\pi\times 200t}] \tag{1.5.15}$$

将它与下式比较：

$$x(t)=\mathrm{Re}[x_l(t)e^{j2\pi\times f_0 t}] \tag{1.5.16}$$

可得：

$$x_l(t)=\mathrm{sinc}(100t) \tag{1.5.17}$$

这意味着在这种情况下，低通等效信号是一个实信号；这样就有 $x_c(t)=x_l(t)$ 和 $x_s(t)=0$，同时也可以得到：

$$\begin{cases} V(t)=|x_c(t)| \\ \Theta(t)=\begin{cases}0, & x_c(t)\geqslant 0\\ \pi, & x_c(t)<0\end{cases}\end{cases} \tag{1.5.18}$$

图 1.28 给出了 $x_c(t)$ 和 $V(t)$ 的图。值得注意的是，选定频率 f_0 是为了在这些图中相对于 f_0 而言 $X(f)$ 是对称的结果。

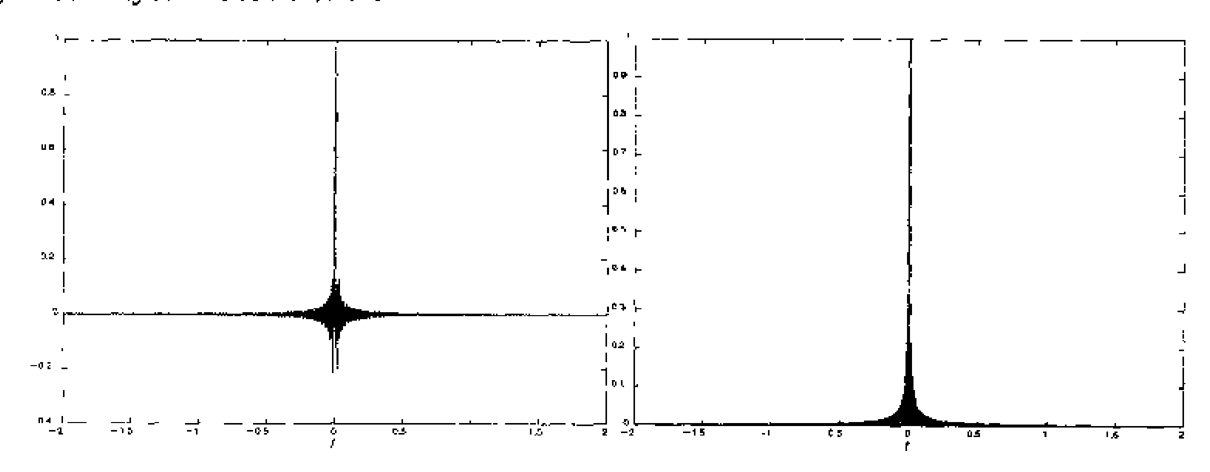


图 1.28　$x(t)$ 的同相分量和包络

3. 如果 $f_0=100$ Hz，那么一般来说上述结果不正确，并且 $x_l(t)$ 会是一个复信号。这时低通等效信号的幅度谱如图 1.29 所示。从这里可见，幅度谱并没有在实信号傅里叶变换中存在的对称性。图 1.30 是 $x(t)$ 的同相分量和它的包络图。

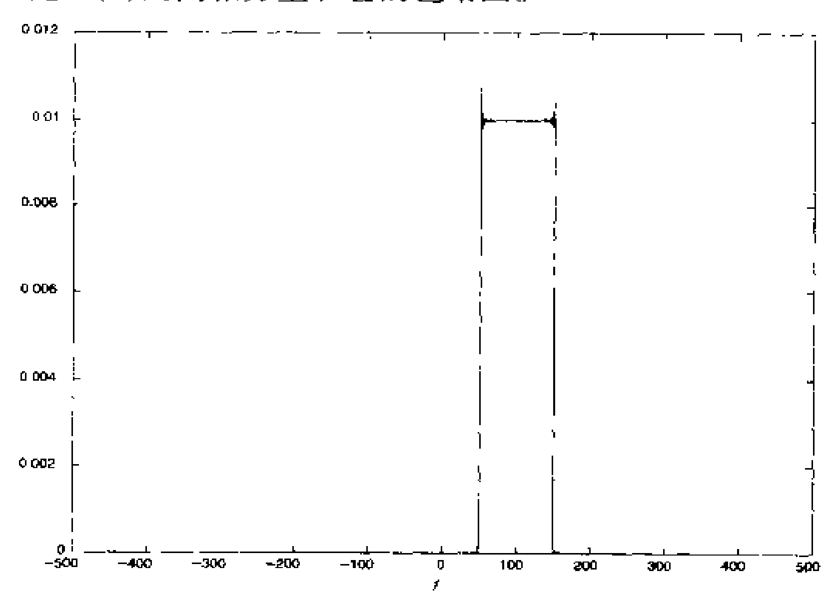


图 1.29　当 $f_0=100$ Hz 时，解说题 1.9 中 $x(t)$ 的低通等效的幅度谱

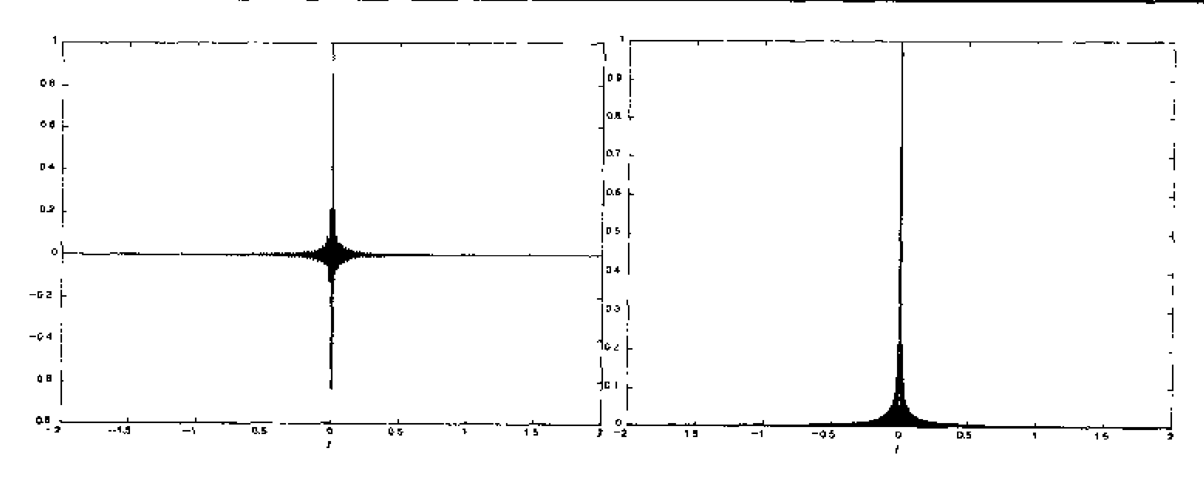


图 1.30　当 $f_0 = 100$ Hz 时，信号 $x(t)$ 的同相分量和包络

1.6　习题

1.1　考虑图 1.1 中展示的解说题 1.1 的周期信号，假设 $A=1$，$T_0=10$，$t_0=1$，求出并画出该信号的离散谱。将结果与解说题 1.1 的结果进行比较，陈述它们的差异。

1.2　在解说题 1.1 中，假设 $A=1$，$T_0=4$，$t_0=2$，求出并画出该信号的离散谱。将结果与解说题 1.1 的结果进行比较，陈述它们的差异。

1.3　利用 m 文件 fseries.m，求图 1.1 中的信号在 $A=1$，$T_0=4$，$t_0=\frac{1}{2}$，并且 $-24 \leqslant n \leqslant 24$ 时的傅里叶级数系数，画出信号的幅度谱。现在再用式(1.25)求傅里叶级数系数并画出幅度谱。为什么这些结果不完全相同？

1.4　用 $T_0=4.6$ 重做习题 1.3，并将结果与用式(1.25)所得结果进行比较。在这两个结果之间能够观察到同样的差异吗？为什么？

1.5　信号 $x(t)$ 在区间 $[-2.3, 2.3]$ 内的关系是 $x(t)=\Lambda(t)$，周期 $T_0=4.6$，利用 MATLAB 脚本 dis-spct.m 求出并画出该周期信号的幅度谱和相位谱，画出在 $-24 \leqslant n \leqslant 24$ 范围内的频谱。现在用解析法求该信号的傅里叶级数系数，并证明所有系数都是非负实数。原先画出的相位谱与这个结果相符吗？如果不相符，为什么？

1.6　将习题 1.5 中的 $x(t)=\Lambda(t)$ 的定义区间改为 $[-1.3, 3.3]$，周期仍为 $T_0=4.6$。利用 MATLAB 脚本 dis-spct.m 求出并画出信号的幅度谱和相位谱。注意，这时的信号与习题 1.5 中的是相同的。请比较这两个习题中的幅度谱和相位谱。幅度谱和相位谱中的哪个表现出了更显著的差异？为什么？

1.7　假设 $[a, b]=[-4, 4]$ 和 $x(t)=\cos(\pi t/8)$，$|t| \leqslant 4$，重做解说题 1.2。

1.8　假设 $[a, b]=[-4, 4]$ 和 $x(t)=\sin(\pi t/8)$，$|t| \leqslant 4$，重做解说题 1.2，并将结果与习题 1.7 的结果进行比较。

1.9　$x(t)$ 的周期为 10^{-6} 并按下式定义：

$$x(t)=\begin{cases}-10^6 t+0.5, & 0 \leqslant t \leqslant 5\times 10^{-7} \\ 0, & \text{其余 } t\end{cases}$$

用数值法求出并画出该信号在 $|t| \leqslant 5\times 10^{-7}$ 区间内的幅度谱和相位谱。

1.10 周期信号 $x(t)$的周期为 $T_0=6$，$x(t)=\Pi(t/3)$，$|t|\leqslant 3$。该信号通过一个冲激响应为

$$h(t)=\begin{cases}e^{-t/2}, & 0\leqslant t\leqslant 4\\ 0 & \text{其余 } t\end{cases}$$

的 LTI 系统，用数值法求出并画出输出信号的离散谱。

1.11 用 $x(t)=e^{-2t}$，$|t|\leqslant 3$ 和

$$h(t)=\begin{cases}1, & 0\leqslant t\leqslant 4\\ 0 & \text{其余 } t\end{cases}$$

重做习题 1.10。

1.12 对于信号 $x(t)=\Pi(t)$和 $y(t)=\Lambda(t)$，利用数值法通过直接求卷积以及通过两信号的傅里叶变换求卷积，验证傅里叶变换卷积定理。

1.13 信号 $x(t)$定义为：

$$x(t)=\begin{cases}1, & -2\leqslant t\leqslant -1\\ |t|, & |t|<1\\ 1, & 1\leqslant t<2\\ 0, & \text{其余 } t\end{cases}$$

画出该信号的幅度谱和相位谱。

1.14 偶信号 $x(t)$对正的 t 值有如下定义：

$$x(t)=\begin{cases}t+1, & 0\leqslant t\leqslant 1\\ 2, & 1\leqslant t\leqslant 2\\ -t+4, & 2\leqslant t\leqslant 4\\ 0, & \text{其余 } t\end{cases}$$

分别用解析法和数值法求出并画出该信号的幅度谱，并比较它们的结果。

1.15 习题 1.14 中的信号通过一个冲激响应为：

$$h(t)=\begin{cases}1, & 0\leqslant t\leqslant 2\\ 2, & 2<t\leqslant 3\\ 0, & \text{其余 } t\end{cases}$$

的 LTI 系统，求输出信号的幅度谱和相位谱。

1.16 现考虑信号 $x(t)$，

$$x(t)=\begin{cases}\cos(2\pi\times 47t)+\cos(2\pi\times 219t), & 0\leqslant t\leqslant 10\\ 0, & \text{其余 } t\end{cases}$$

和解说题 1.8 中一样，假设这个信号按 1000 样本/s 的采样率进行采样。利用 m 文件 butter.m 设计一个截止频率为 100 Hz 的 4 阶巴特沃斯低通滤波器，并将信号 $x(t)$通过该滤波器，求出并概略画出输出的功率谱，将结果与图 1.25 进行比较。再设计一个具有相同截止频率的 8 阶巴特沃斯低通滤波器，求该滤波器的输出，画出它的功率谱，试比较这两种情况下的结果。

1.17 重做习题 1.16，但这次设计的是具有相同阶和截止频率的巴特沃斯高通滤波器。画出结果并进行比较。

1.18 考虑信号

$$x(t)=\begin{cases}\cos(2\pi\times 47t)+\cos(2\pi\times 219t), & 0\leqslant t\leqslant 10\\ 0, & \text{其余 } t\end{cases}$$

a. 求对应于该信号的解析信号。

b. 求出并画出这个信号的希尔伯特变换。

c. 求出并画出这个信号的包络。

d. 分别假设 $f_0=47$ Hz 和 $f_0=219$ Hz，求该信号的低通等效以及同相和正交分量。

第2章 随机过程

2.1 概述

这一章介绍了产生随机变量和随机过程样本的方法。首先从描述产生给定概率分布函数的随机变量的方法入手，然后考虑高斯(Gauss)和高斯－马尔可夫(Gauss-Markov)过程，并说明产生这些过程的方法。我们讨论的第三个问题是在时域中用自相关函数和在频域中用功率谱描述的平稳随机过程的特性。由于线性滤波器在通信系统中起着非常重要的作用，所以还要讨论经由线性过滤后的随机过程的自相关函数和功率谱。本章最后一节讲述处理低通和带通型随机过程的特性。

2.2 随机变量的产生

在实际情况下常常用随机数发生器来仿真类似于噪声信号的效果以及物理世界中的其他随机现象。在电子器件和系统中存在着这类噪声，而且这些噪声通常限制了远距离通信和检测相对微弱的信号的能力。利用在计算机上产生的这类噪声，就有可能通过通信系统的仿真来研究噪声的影响，并估计噪声存在时系统的性能。

大多数计算机的软件库中都包含一个均匀随机数发生器，它以等概率产生 0 和 1 之间的一个数。我们称这样的随机数发生器的输出为**随机变量**。如果将这个随机变量记为 A，它的大小范围则为 $0\leqslant A\leqslant 1$。

我们知道，一台数字计算机的数值输出具有有限精度，这样就不可能在区间 $0\leqslant A\leqslant 1$ 内表示数的连续值。然而，可以假定计算机用了一个很大的比特数来表示每个输出；既可以采用定点制，也可以采用浮点制。结果，从实际的角度来说，可以认为在范围 $0\leqslant A\leqslant 1$ 内输出的数位是足够大的，以至于在这个范围内的任何值都可能从发生器输出。

随机变量 A 的均匀概率密度函数 $f(A)$ 如图 2.1(a)所示，注意 A 的平均值(或均值) m_A 是 1/2。概率密度函数的积分代表 $f(A)$ 下的面积，称为随机变量 A 的**概率分布函数**，定义为：

$$F(A)=\int_{-\infty}^{A}f(x)\mathrm{d}x \tag{2.2.1}$$

对任何随机变量而言，这个面积一定总是 1，它是由分布函数能实现的最大值。所以，对于均匀随机变量 A 来说，则有

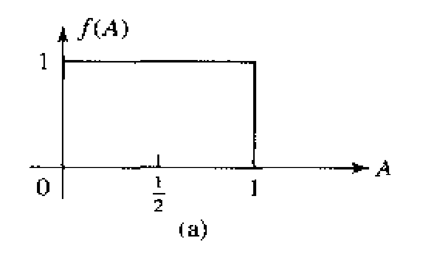
$$F(1)=\int_{-\infty}^{1}f(x)\mathrm{d}x=1 \tag{2.2.2}$$

$F(A)$ 的范围就是 $0\leqslant F(A)\leqslant 1, 0\leqslant A\leqslant 1$。概率分布函数如图 2.1(b)所示。

如果想产生一个在 $(b, b+1)$ 内的均匀分布噪声，只需要在随机数发生器输出 A 上偏置一个 b 量即可实现。据此，新随机变量 B 就能定义为：

$$B=A+b \tag{2.2.3}$$

而现在的均值 $m_B = b + \frac{1}{2}$。例如，若 $b = -\frac{1}{2}$，则这个随机变量 B 就在 $\left(-\frac{1}{2}, \frac{1}{2}\right)$ 内均匀分布。如图 2.2(a)所示，它的概率分布函数 $F(B)$ 如图 2.2(b)所示。



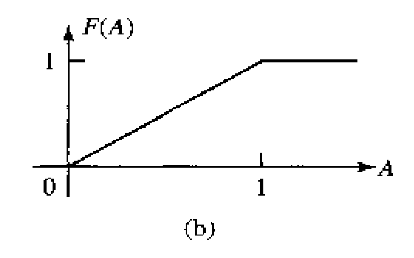
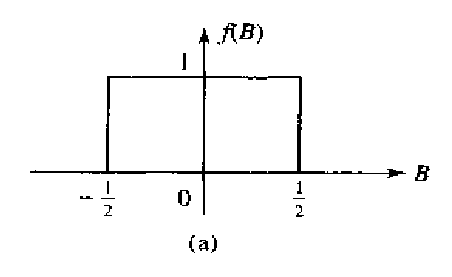


图 2.1 均匀分布随机变量 A 的概率密度函数 $f(A)$ 和概率分布函数 $F(A)$



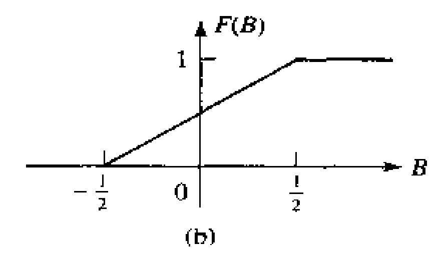


图 2.2 一个零均值均匀分布随机变量的概率密度函数和概率分布函数

在(0,1)内的均匀分布随机变量可用来产生具有其他概率分布函数的随机变量。例如，设想要产生一个随机变量 C，其概率分布函数 $F(C)$ 如图 2.3 所示。

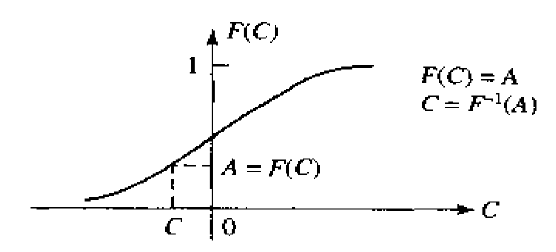


图 2.3 从均匀分布随机变量 A 到新随机变量 C 的逆映射

因为 $F(C)$ 的范围在(0,1)内，可以先产生一个在(0,1)内的均匀分布随机变量 A。若置

$$F(C) = A \tag{2.2.4}$$

则有：

$$C = F^{-1}(A) \tag{2.2.5}$$

由此，对 C 解出式(2.2.4)和式(2.2.5)的解，就给出了具有 $F(C) = A$ 的 C 值。按照这一方法，就求得了具有概率分布函数 $F(C)$ 的新随机变量 C。这种从 A 到 C 的逆映射如图 2.3 所示。

解说题

解说题 2.1

产生一个随机变量 C，它具有图 2.4(a)所示的线性概率密度函数，即

$$f(C) = \begin{cases} \frac{1}{2}C, & 0 \leqslant C \leqslant 2 \\ 0, & \text{其余 } C \end{cases}$$

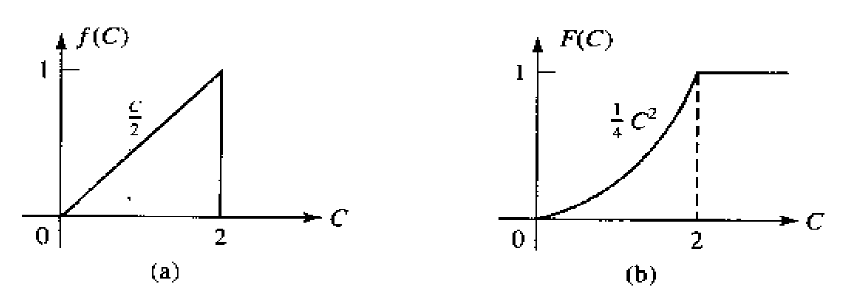


图 2.4 线性概率密度函数和对应的概率分布函数

题 解

该随机变量的概率分布函数为：

$$F(C)=\begin{cases}0, & C<0\\ \frac{1}{4}C^2, & 0\leqslant C\leqslant 2\\ 1, & C>2\end{cases}$$

如图 2.4(b)所示。现在，产生一个均匀分布的随机变量 A，并置 $F(C)=A$，所以有：

$$F(C)=\frac{1}{4}C^2=A \tag{2.2.6}$$

解出 C 为：

$$C=2\sqrt{A} \tag{2.2.7}$$

由此，产生的具有概率分布函数 $F(C)$ 的随机变量 C 如图 2.4(b)所示。

在解说题 2.1 中，这种逆映射 $C=F^{-1}(A)$ 很简单；而在有些情况中可没有这么简单。试图产生一个具有正态分布函数的随机数就属于这种情况。

在实际系统中所遇到的噪声往往是由正态(或高斯)概率分布来表征的，这个分布如图 2.5所示。这个概率密度函数由下式给出：

$$f(C)=\frac{1}{\sqrt{2\pi}\sigma}e^{-C^2/(2\sigma^2)},\quad -\infty<C<\infty \tag{2.2.8}$$

其中 σ^2 是 C 的方差，它是概率密度函数 $f(C)$ 的分散程度的一种度量。概率分布函数 $F(C)$ 是在区间 $(-\infty, C)$ 内 $f(C)$ 下所包围的面积，即



$$F(C)=\int_{-\infty}^{C}f(x)\mathrm{d}x \tag{2.2.9}$$

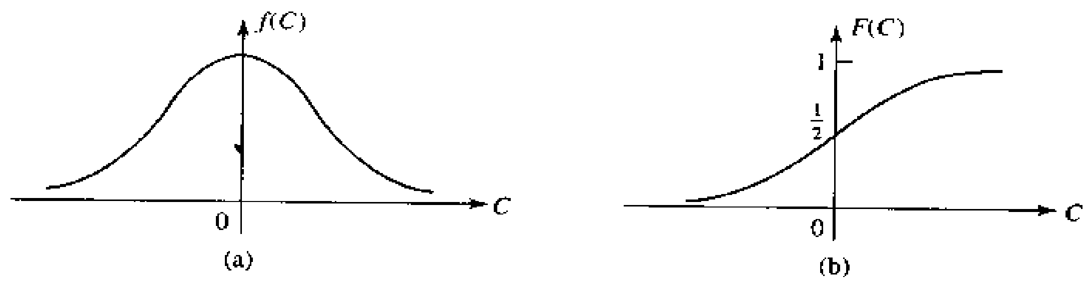


图 2.5 高斯概率密度函数和对应的概率分布函数

遗憾的是，式(2.2.9)的积分无法用简单的函数来表示，使得完成逆映射很困难。已经找

到了克服这个难题的一种办法。由概率论知道,具有概率分布函数为:

$$F(R)=\begin{cases}0, & R<0\\ 1-\mathrm{e}^{-R^2/(2\sigma^2)}, & R\geqslant 0\end{cases} \tag{2.2.10}$$

的瑞利分布随机变量 R 与一对高斯随机变量 C 和 D 是通过如下变换:

$$C=R\cos\Theta \tag{2.2.11}$$

$$D=R\sin\Theta \tag{2.2.12}$$

关联的。这里 Θ 是在$(0,2\pi)$内的均匀分布变量,参数 σ^2 是 C 和 D 的方差。因为式(2.2.10)容易求得逆函数,所以:

$$F(R)=1-\mathrm{e}^{-R^2/(2\sigma^2)}=A \tag{2.2.13}$$

并且

$$R=\sqrt{2\sigma^2\ln\left(\frac{1}{1-A}\right)} \tag{2.2.14}$$

其中 A 是在$(0,1)$内均匀分布的随机变量。现在,如果我们产生了第二个均匀分布的随机变量 B,而定义:

$$\Theta=2\pi B \tag{2.2.15}$$

那么,从式(2.2.11)和式(2.2.12)可求得两个统计独立的高斯分布随机变量 C 和 D。

上面所述的方法在实际中常被作为产生高斯分布的随机变量。正如在图 2.5 中所看到的,这些随机变量有一个零均值和方差 σ^2。如果想要一个非零均值的高斯随机变量,那么用加一个均值的办法将变量 C 和 D 进行转换即可。

实现前述产生高斯分布随机变量方法的 MATLAB 脚本如下所示。

m 文件

```
function [gsrv1,gsrv2]=gngauss(m,sgma)
%    [gsrv1,gsrv2]=gngauss(m,sgma)
%    [gsrv1,gsrv2]=gngauss(sgma)
%    [gsrv1,gsrv2]=gngauss
%              GNGAUSS  generates two independent Gaussian random variables with mean
%              m and standard deviation sgma. If one of the input arguments is missing,
%              it takes the mean as 0.
%              If neither the mean nor the variance is given, it generates two standard
%              Gaussian random variables.
if nargin == 0,
  m=0; sgma=1;
elseif nargin == 1,
  sgma=m; m=0;
end;
u=rand;                                   % a uniform random variable in (0,1)
z=sgma*(sqrt(2*log(1/(1-u))));            % a Rayleigh distributed random variable
u=rand;                                   % another uniform random variable in (0,1)
gsrv1=m+z*cos(2*pi*u);
gsrv2=m+z*sin(2*pi*u);
```

2.3　高斯和高斯-马尔可夫过程

在通信系统中,高斯过程起着非常重要的作用,其最根本的原因是在电子器件中的热噪声

(由热骚动引起电子的随机运动而产生)能够用一个高斯过程准确地建模。对于热噪声具有高斯行为的解释是:电路中由电子运动引起的电流可以看成是大量的小电流(即单个电子)的相加。可以假设,至少这些源中的大多数在特性行为上是独立的。因此,总电流就是这些大量的独立和同分布的随机变量之和。根据中心极限定理,总电流就有一个高斯分布。

除了热噪声以外,高斯过程也对其他一些信息源给出了相当好的模型。下面要给出的有关高斯过程的一些性质,使得这些过程在教学上也很容易处理。现在给出一个正规的高斯过程定义。

定义 若对全部 n 和全部$(t_1,t_2,\cdots,t_n)$,随机变量$\{X(t_i)\}_{i=1}^{n}$具有联合高斯密度函数,这个随机过程 $X(t)$就是高斯过程,可以表示成:

$$f(\boldsymbol{x}) = \frac{1}{(2\pi)^{n/2}[\det(\boldsymbol{C})]^{1/2}}\exp\left[-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{m})^{\mathrm{t}}\boldsymbol{C}^{-1}(\boldsymbol{x}-\boldsymbol{m})\right] \tag{2.3.1}$$

其中向量 $\boldsymbol{x}=(x_1,x_2,\cdots,x_n)^{\mathrm{T}}$ 记为 n 个随机变量 $x_i \equiv X(t_i)$,$\boldsymbol{m}$ 是均值向量,即 $\boldsymbol{m}=E(\boldsymbol{X})$,$\boldsymbol{C}$ 是随机变量$(x_1,x_2,\cdots,x_n)$的 $n\times n$ 协方差矩阵,该矩阵的元素是:

$$c_{ij} = E[(x_i - m_i)(x_j - m_j)] \tag{2.3.2}$$

上角标 T 记为向量或矩阵的转置,$\boldsymbol{C}^{-1}$是协方差矩阵 $\boldsymbol{C}$ 的逆矩阵。

从这个定义尤其可见,在任意瞬时 t_0,随机变量 $X(t_0)$是高斯型的;在任意两点 t_1 和 t_2 上,随机变量 $X(t_1)$和 $X(t_2)$是按照二维高斯随机变量分布的。甚至,因为对$\{X(t_i)\}_{i=1}^{n}$的完全统计描述仅决定于均值向量 $\boldsymbol{m}$ 和协方差矩阵 $\boldsymbol{C}$,所以有下面的性质:

性质 1 对于高斯过程,均值 $\boldsymbol{m}$ 和协方差矩阵 $\boldsymbol{C}$ 给出了对过程的完全统计描述。

高斯过程的另一个很重要的性质是当它通过一个线性时不变(LTI)系统时所具有的特性,这个性质可陈述如下:

性质 2 如果高斯过程 $X(t)$通过一个线性时不变(LTI)系统,其输出也是一个高斯过程。系统对 $X(t)$的作用只是影响 $X(t)$的均值和协方差的改变。

解说题

解说题 2.2 [多变量高斯过程样本的产生]

产生具有给定均值 $\boldsymbol{m}_x$ 和协方差 $\boldsymbol{C}_x$ 的多变量高斯过程 $X(t)$的样本。

题 解

首先,用 2.2 节给出的方法产生一个统计独立的零均值和单位方差的高斯随机变量的序列 n,用向量 $\boldsymbol{Y}=(y_1,y_2,\cdots,y_n)^{\mathrm{T}}$ 表示这 n 个样本的序列。其次,将这个期望的 $n\times n$ 协方差矩阵 $\boldsymbol{C}_x$ 分解为:

$$\boldsymbol{C}_x = \boldsymbol{C}_x^{1/2}(\boldsymbol{C}_x^{1/2})^{\mathrm{t}} \tag{2.3.3}$$

然后定义线性变换的$(n\times 1)$向量 $\boldsymbol{X}$ 为:

$$\boldsymbol{X} = \boldsymbol{C}_x^{1/2}\boldsymbol{Y} + \boldsymbol{m}_x \tag{2.3.4}$$

由此,$\boldsymbol{X}$ 的协方差是:

$$\boldsymbol{C}_x = E[(\boldsymbol{X}-\boldsymbol{m}_x)(\boldsymbol{X}-\boldsymbol{m}_x)^{\mathrm{T}}]$$

$$= E[C_x^{1/2}YY'(C_x^{1/2})']$$
$$= C_x^{1/2}E(YY')(C_x^{1/2})'$$
$$= C_x^{1/2}(C_x^{1/2})' \tag{2.3.5}$$

在这个过程中,最困难的一步就是协方差矩阵 C_x 的分解。现在用一个采用双变量高斯分布的例子来阐明这个过程。设想开始时有一对统计独立的高斯随机变量 y_1 和 y_2,它们都具有零均值和单位方差。我们要将这一对随机变量转换到均值 $m=0$ 并且协方差矩阵为:

$$C = \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix} = \begin{bmatrix} 1 & \frac{1}{2} \\ \frac{1}{2} & 1 \end{bmatrix} \tag{2.3.6}$$

的一对高斯随机变量 x_1 和 x_2。式中 σ_1^2 和 σ_2^2 分别是 x_1 和 x_2 的方差,而 ρ 是归一化方差,定义为:

$$\rho = \frac{E[(X_1 - m_1)(X_2 - m_2)]}{\sigma_1\sigma_2} = \frac{c_{12}}{\sigma_1\sigma_2} \tag{2.3.7}$$

这个协方差矩阵 C 可以分解为:

$$C = C^{1/2}(C^{1/2})'$$

其中,

$$C^{1/2} = \frac{1}{2\sqrt{2}}\begin{bmatrix} \sqrt{3}+1 & \sqrt{3}-1 \\ \sqrt{3}-1 & \sqrt{3}+1 \end{bmatrix} \tag{2.3.8}$$

因此,

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = C^{1/2}\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$
$$= \frac{1}{2\sqrt{2}}\begin{bmatrix} \sqrt{3}+1 & \sqrt{3}-1 \\ \sqrt{3}-1 & \sqrt{3}+1 \end{bmatrix}\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$
$$= \frac{1}{2\sqrt{2}}\begin{bmatrix} (\sqrt{3}+1)y_1 + (\sqrt{3}-1)y_2 \\ (\sqrt{3}-1)y_1 + (\sqrt{3}+1)y_2 \end{bmatrix} \tag{2.3.9}$$

实现这个计算的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 2.2.
echo on
mx=[0 0]';
Cx=[1 1/2;1/2 1];
x=multi_gp(mx,Cx);
% Computation of the pdf of (x1,x2) follows.
delta=0.3;
x1=-3:delta:3;
x2=-3:delta:3;
for i=1:length(x1),
   for j=1:length(x2),
      f(i,j)=(1/((2*pi)*det(Cx)^1/2))*exp((-1/2)*(([x1(i) x2(j)]-mx')*inv(Cx)*([x1(i);x2(j)]-mx)));
      echo off ;
   end;
```

```
end;
echo on ;
% Plotting command for pdf follows.
mesh(x1,x2,f);
```

m 文件

```
function [x] = multi_gp(m,C)
%   [x]=multi_gp(m,C)
%                  MULTI_GP   generates a multivariate Gaussian random
%                  process with mean vector m (column vector) and covariance matrix C.
N=length(m);
for i=1:N,
  y(i)=gngauss;
end;
y=y.' ;
x=sqrtm(C)*y+m;
```

图 2.6 说明了由式(2.3.6)给出的协方差矩阵 $\boldsymbol{C}$ 的联合概率密度函数 $f(x_1,x_2)$。

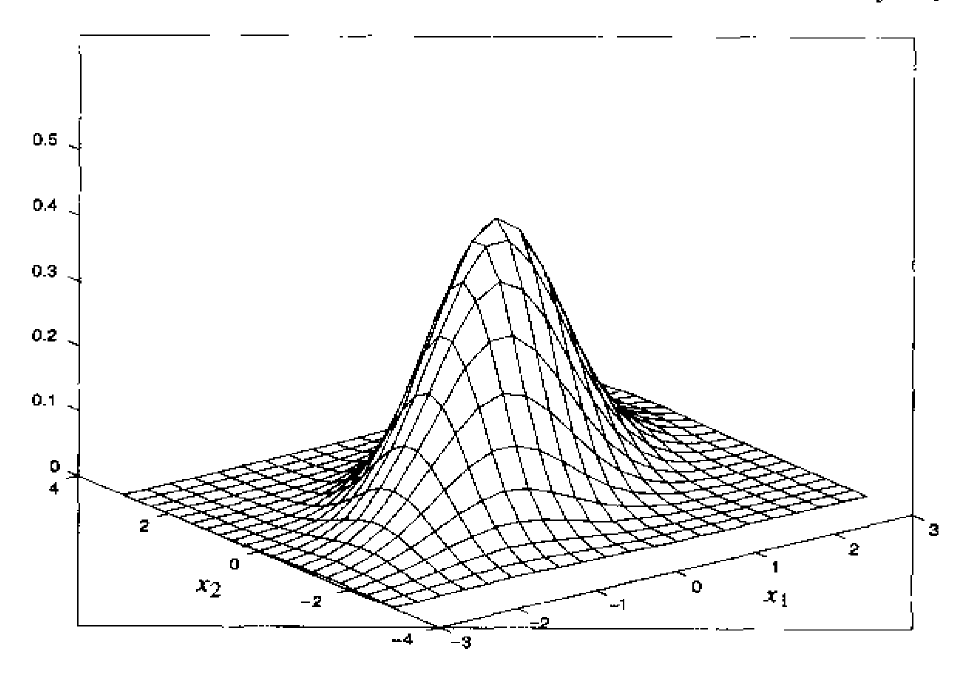


图 2.6　x_1 和 x_2 的联合概率密度函数

正如前面指出的，在计算中最困难的一步是求 $\boldsymbol{C}^{1/2}$。已知所要求的协方差矩阵，就可以求出特征值 $\{\lambda_k, 1 \leqslant k \leqslant n\}$ 和相应的特征向量 $\{\boldsymbol{v}_k, 1 \leqslant k \leqslant n\}$，然后协方差矩阵 $\boldsymbol{C}$ 可以表示为：

$$\boldsymbol{C} = \sum_{k=1}^{n} \lambda_k \boldsymbol{v}_k \boldsymbol{v}_k^{t} \tag{2.3.10}$$

又因为 $\boldsymbol{C} = \boldsymbol{C}^{1/2}(\boldsymbol{C}^{1/2})^{t}$，可得：

$$\boldsymbol{C}^{1/2} = \sum_{k=1}^{n} \boldsymbol{\lambda}_k^{1/2} \boldsymbol{v}_k \boldsymbol{v}_k^{t} \tag{2.3.11}$$

定义 一个**马尔可夫过程** $X(t)$是一个随机过程，如果该过程的当前状况是给定的，那么过程的过去对将来没有任何影响；也就是说，若 $t_n > t_{n-1}$，则

$$P[X(t_n) \leqslant x_n \mid X(t), t \leqslant t_{n-1}] = P[X(t_n) \leqslant x_n \mid X(t_{n-1})] \quad (2.3.12)$$

根据这个定义，应该得出：若 $t_1 < t_2 < \cdots < t_n$，则

$$P[X(t_n) \leqslant x_n \mid X(t_{n-1}), X(t_{n-2}), \cdots, X(t_1)] = P[X(t_n) \leqslant x_n \mid X(t_{n-1})] \quad (2.3.13)$$

定义 一个**高斯-马尔可夫过程** $X(t)$是一个概率密度函数为高斯型的马尔可夫过程。

产生马尔可夫过程的最简单方法是利用下面这个简单的递推公式：

$$X_n = \rho X_{n-1} + w_n \quad (2.3.14)$$

w_n 是一个零均值、独立和同分布的（白色）随机变量，ρ 是确定 X_n 和 X_{n-1}之间相关程度的一个参数，即

$$E(X_n X_{n-1}) = \rho E(X_{n-1}^2) = \rho \sigma_{n-1}^2 \quad (2.3.15)$$

如果序列$\{w_n\}$是高斯型的，那么所得到的过程 $X(t)$就是一个高斯-马尔可夫过程。

解说题

解说题 2.3 ［高斯-马尔可夫过程］

用递推关系：

$$X_n = 0.95X_{n-1} + w_n, \qquad n = 1,2,\cdots,1000 \quad (2.3.16)$$

产生1000个样本（等间隔）的高斯-马尔可夫过程序列，这里 $X_0 = 0$，$\{w_n\}$是零均值、单位方差的独立同分布的高斯随机变量。作为时间变量 n 的函数，画出这个序列$\{X_n, 1 \leqslant n \leqslant 1000\}$和自相关函数：

$$\hat{R}_x(m) = \frac{1}{N-m}\sum_{n=1}^{N-m} X_n X_{n+m}, \quad m = 0,1,\cdots,50 \quad (2.3.17)$$

其中 $N = 1000$。

题解

关于上述计算的MATLAB脚本如下所示。图2.7和图2.8分别为序列(X_n)和自相关函数$\hat{R}(m)$。

m文件

```
% MATLAB script for Illustrative Problem 2.3.
echo on
rho=0.95;
X0=0;
N=1000;
X=gaus_mar(X0,rho,N);
M=50;
Rx=Rx_est(X,M);
% Plotting commands follow.
```

m文件

```
function [X]=gaus_mar(X0,rho,N)
% [X]=gaus_mar(X0,rho,N)
%             GAUS_MAR   generates a Gauss-Markov process of length N.
```

```
%                The noise process is taken to be white Gaussian
%                noise with zero mean and unit variance.
for i=1:2:N,
  [Ws(i) Ws(i+1)]=gngauss;          % Generate the noise process.
end;
X(1)=rho*X0+Ws(1);                   % first element in the Gauss-Markov process
for i=2:N,
  X(i)=rho*X(i-1)+Ws(i);            % the remaining elements
end;
```

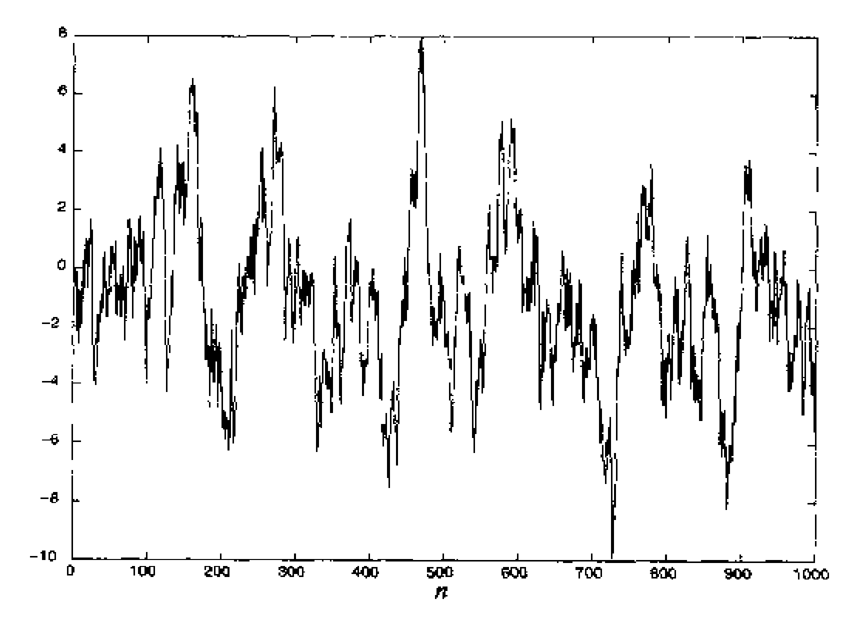


图 2.7 高斯-马尔可夫序列

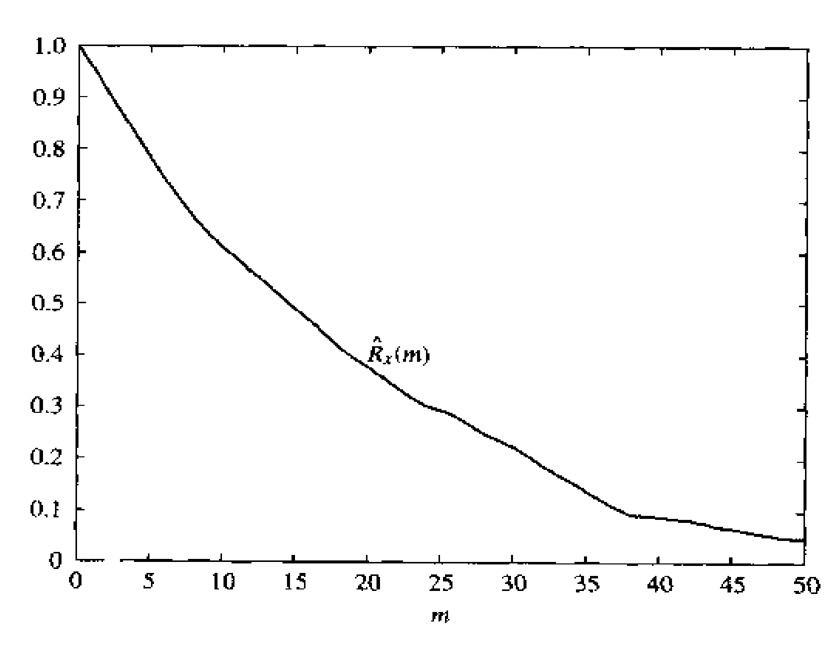


图 2.8 高斯-马尔可夫过程的自相关函数

2.4 随机过程的功率谱和白色过程

一个平稳随机过程 $X(t)$在频域中是用它的功率谱$\mathcal{S}_x(f)$来表征的，功率谱是随机过程自相关函数 $R_x(\tau)$的傅里叶变换，即

$$\mathcal{S}_x(f) = \int_{-\infty}^{\infty} R_x(\tau)e^{-j2\pi f\tau}d\tau \tag{2.4.1}$$

相反，一个平稳随机过程 $X(t)$的自相关函数 $R_x(\tau)$可以由功率谱$\mathcal{S}_x(f)$的傅里叶逆变换得到，即

$$R_x(\tau) = \int_{-\infty}^{\infty} \mathcal{S}_x(f)e^{j2\pi f\tau}df \tag{2.4.2}$$

对实现通信系统时使用的电子器件所产生的热噪声进行建模的时候，往往假设这样的噪声是一个**白色随机过程**。这个过程定义如下：

定义　若一个随机过程具有平坦的功率谱，即若$\mathcal{S}_x(f)$对全部 f 是一个常数，则称该随机过程为**白色过程**。

正如上面已指出的，白色过程的重要性正是来自于热噪声能在很宽的频率范围内作为一个常数谱准确地建模。另外，用于描述各种信息源的许多过程都能仿真为在白色过程驱动下的 LTI 系统的输出。

然而我们可以看到，若对所有的 f，$\mathcal{S}_x(f) = C$，那么

$$\int_{-\infty}^{\infty} \mathcal{S}_x(f)df = \int_{-\infty}^{\infty} Cdf = \infty \tag{2.4.3}$$

这样，总功率是无限大的。很显然，没有一个真实的物理过程能有无限大的功率，因此白色过程不可能是一个有意义的物理过程。然而，热噪声的量子力学分析证明，它的功率谱密度为：

$$\mathcal{S}_n(f) = \frac{hf}{2(e^{hf/(kT)} - 1)} \tag{2.4.4}$$

其中 h 为普朗克常数(等于 6.6×10^{-34} J·s)，k 是玻尔兹曼常数(等于 1.38×10^{-23} J/K)，T 为热力学温度，这个功率谱如图 2.9 所示。

这个谱在 $f=0$ 时为最大值，其值为 $kT/2$。随着 f 趋于无限大，谱趋近于 0；但是，收敛到 0 的速度非常慢。例如，在室温下($T = 300$ K)，大约在 $f\approx2\times10^{12}$ Hz 时，$\mathcal{S}_n(f)$才降到它的最大值的 90%，而这个频率已大大超过在通常通信系统中所使用的频率。从这里可以得出，虽然热噪声并不是真正为白色的，但是从实际的角度来看，热噪声还是可以把它当成功率谱等于 $kT/2$ 的白色过程来建模，kT 值通常记为 N_0。因此，热噪声的功率谱密度一般就由$\mathcal{S}_n(f) = N_0/2$ 给出，有时将它称为**双边功率谱密度**，以强调这个谱既伸展到正频率，又伸展到负频率。在本书中不使用这个术语，只使用**功率谱**或**功率谱密度**。

对于白色随机过程 $X(t)$，其功率谱$\mathcal{S}_x(f) = N_0/2$，自相关函数 $R_x(\tau)$就是：

$$R_x(\tau) = \int_{-\infty}^{\infty} \mathcal{S}_x(f)e^{j2\pi f\tau}df = \frac{N_0}{2}\int_{-\infty}^{\infty} e^{j2\pi f\tau}df = \frac{N_0}{2}\delta(\tau) \tag{2.4.5}$$

其中 $\delta(\tau)$为单位冲激函数。这样，只要 $\tau\neq0$，就有 $R_x(\tau)=0$。这就是说，如果对一个白色过程在两个点 t_1 和 $t_2(t_1\neq t_2)$上进行采样，所得到的随机变量一定不相关。如果除了是白色的

以外,随机过程还是高斯分布的,那么已采样的随机变量一定是统计独立的高斯随机变量。

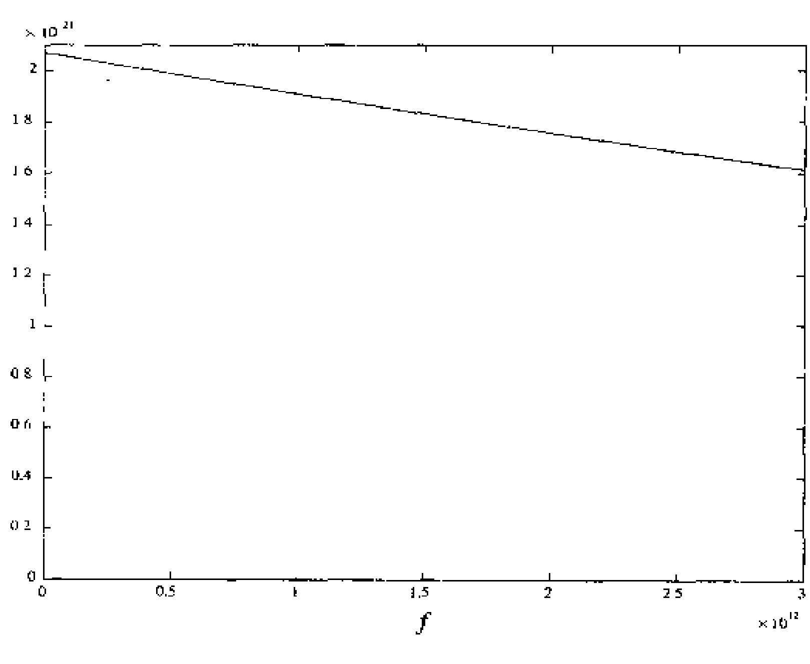


图 2.9 式(2.4.4)中的$S_n(f)$的图

解说题

解说题 2.4 [自相关与功率谱]

在$\left(-\frac{1}{2},\frac{1}{2}\right)$内产生一个 $N=1000$ 的独立同分布的均匀分布随机数的离散时间序列,计算该序列$\{X_n\}$的自相关估值,它定义为:

$$\begin{aligned}\hat{R}_x(m) &= \frac{1}{N-m}\sum_{n=1}^{N-m} X_n X_{n+m}, \quad m = 0,1,\cdots,M \\ &= \frac{1}{N-|m|}\sum_{n=|m|}^{N} X_n X_{n+m}, \quad m = -1,-2,\cdots,-M\end{aligned} \tag{2.4.6}$$

同时,用计算 $\hat{R}_x(m)$的离散傅里叶变换(DFT)求序列$\{X_n\}$的功率谱。DFT 定义为:

$$S_x(f) = \sum_{m=-M}^{M}\hat{R}_x(m)e^{-j2\pi fm/(2(M+1))} \tag{2.4.7}$$

可以用快速傅里叶变换(FFT)算法高效地计算得到。

题 解

下面给出了产生序列$\{X_n\}$,并计算自相关和功率谱$S_x(f)$的 MATLAB 脚本。值得注意的是,自相关函数估值和功率谱都呈现出显著的波动,因此有必要在几次实现上对样本的自相关求平均。运行这个程序,在随机过程的 10 次实现上采用平均自相关得到的 $\hat{R}_x(m)$和$\hat{S}_x(f)$如图 2.10和图 2.11 所示。

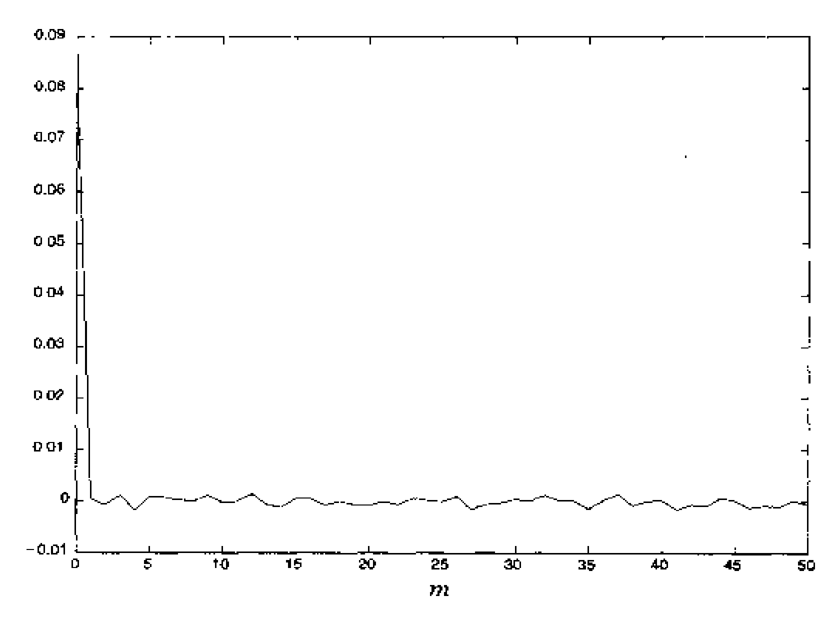


图2.10 解说题2.4中的自相关函数

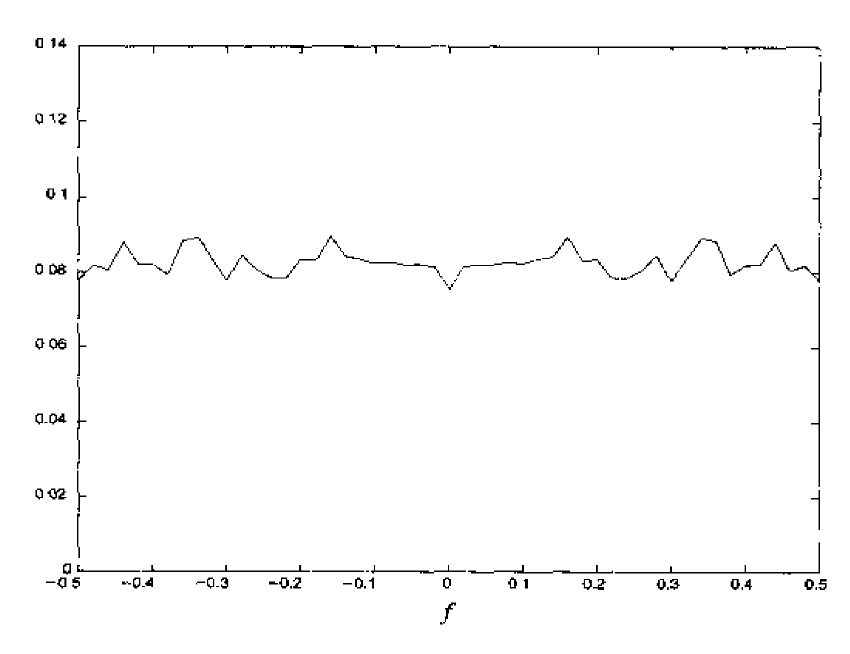


图2.11 解说题2.4中的功率谱

m文件

```
% MATLAB script for Illustrative Problem 2.4.
echo on
N=1000;
M=50;
Rx_av=zeros(1,M+1);
Sx_av=zeros(1,M+1);
for j=1:10,                 % Take the ensemble average over ten realizations
   X=rand(1,N)-1/2;         % N i.i.d. uniformly distributed random variables
                            % between -1/2 and 1/2.
   Rx=Rx_est(X,M);              % autocorrelation of the realization
```

```
    Sx=fftshift(abs(fft(Rx)));        % power spectrum of the realization
    Rx_av=Rx_av+Rx;                   % sum of the autocorrelations
    Sx_av=Sx_av+Sx;                   % sum of the spectrums
    echo off ;
end;
echo on ;
Rx_av=Rx_av/10;                       % ensemble average autocorrelation
Sx_av=Sx_av/10;                       % ensemble average spectrum
% Plotting comments follow
```

一个带限随机过程 $X(t)$的功率谱为：

$$\mathcal{S}_x(f) = \begin{cases} \frac{N_0}{2}, & |f| \leqslant B \\ 0, & |f| > B \end{cases} \tag{2.4.8}$$

现在来求它的自相关函数。由式(2.4.1)有：

$$\begin{aligned} R_x(\tau) &= \int_{-B}^{B} \frac{N_0}{2} e^{j2\pi f\tau} df \\ &= N_0 B\left(\frac{\sin(2\pi B\tau)}{2\pi B\tau}\right) \end{aligned} \tag{2.4.9}$$

$R_x(\tau)$如图 2.12 所示，图中 $R_x(0)$已归一化到 1。

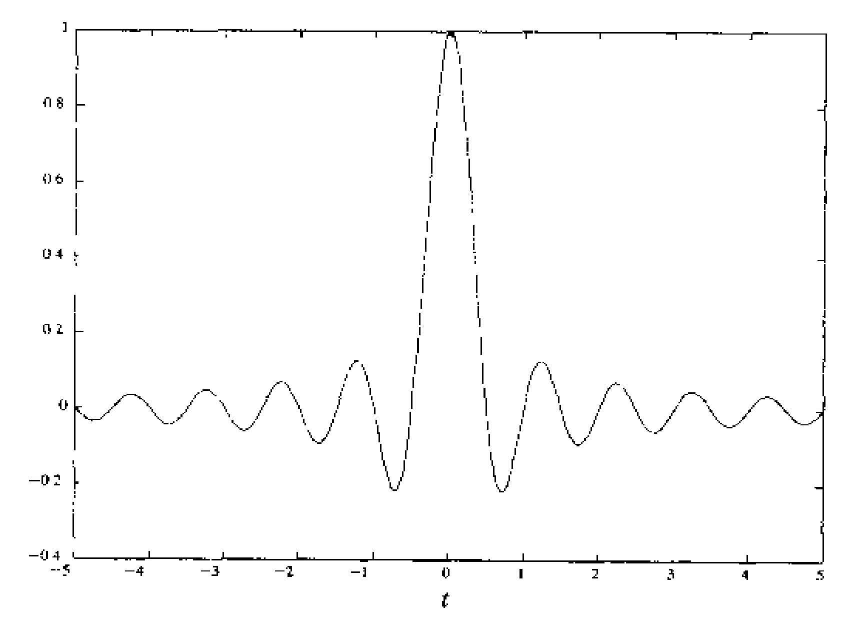


图 2.12 当 $B = N_0 = 1$ 时由式(2.4.9)给出的自相关函数 $R_x(\tau)$的图

MATLAB 可以用于从$\mathcal{S}_x(f)$计算出 $R_x(\tau)$，反之亦然。快速傅里叶变换(FFT)算法可用于这类计算。

解说题

解说题 2.5 [自相关与功率谱]

计算由式(2.4.8)给出功率谱的随机过程的自相关函数 $R_x(\tau)$。

题 解

为了进行计算，用在频率范围 $|f| \leqslant B$ 内的 N 个样本代表 $S_x(f)$，每个样本归一化到 1。图 2.13显示了 $N=32$ 的逆 FFT 计算结果。要注意的是，由于采样 $S_x(f)$ 是仅在 $|f| \leqslant B$ 内进行的，所以得到的仅是自相关函数 $R_x(\tau)$ 的粗略表示。在这个例子中频率间隔是 $\Delta f = 2B/N$。如果将 Δf 保持固定而增加样本数，以包括 $|f| > B$ 的样本，就可以得到 $R_x(\tau)$ 的中间值。图 2.14 显示的是用 $N_1 = 256$ 个样本计算逆 FFT 的结果，图中的这些中间值在 $N=32$ 时都为零[①]。

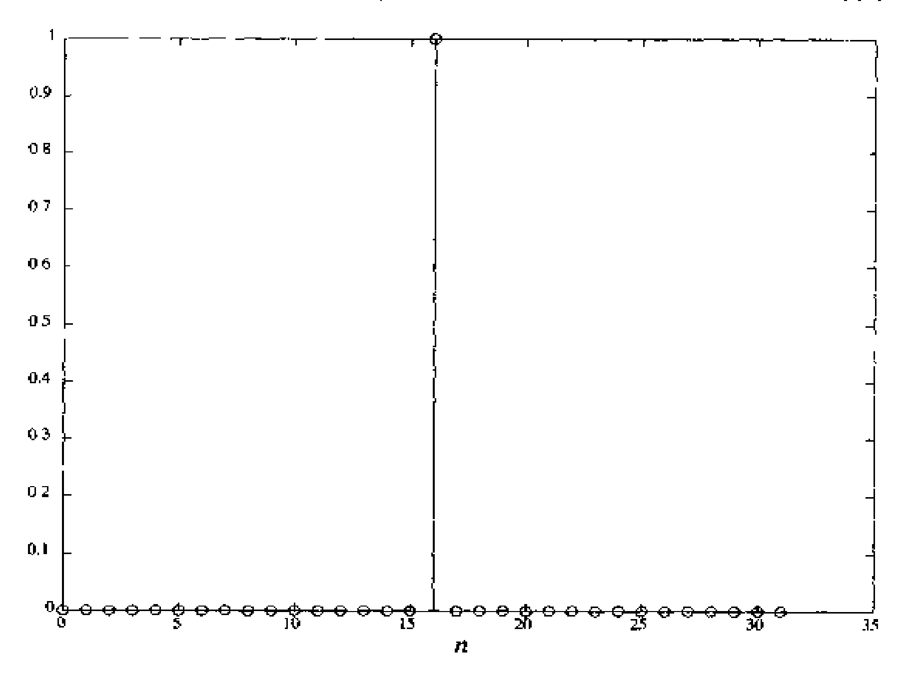


图 2.13 解说题 2.5 中 32 个样本的该带限随机过程功率谱的逆 FFT

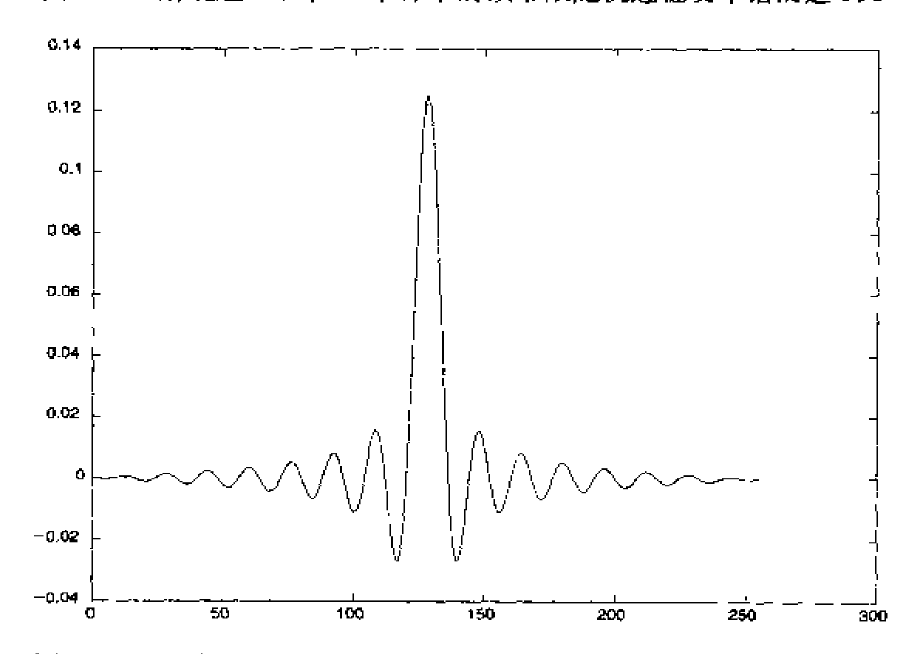


图 2.14 解说题 2.5 中 256 个样本的该带限随机过程功率谱的逆 FFT

① 此处原始为"……都为 1"，似有误。——译者注

2.5 随机过程的线性滤波

假设一个平稳随机过程 $X(t)$通过某个线性时不变滤波器,该滤波器在时域中用它的冲激响应来表征,而在频域中则用它的频率响应

$$H(f)=\int_{-\infty}^{\infty}h(t)e^{-j2\pi ft}dt \tag{2.5.1}$$

来表征,这样可得到该线性滤波器的输出是随机过程:

$$Y(t)=\int_{-\infty}^{\infty}X(t)h(t-\tau)d\tau \tag{2.5.2}$$

$Y(t)$的均值是:

$$\begin{aligned}m_Y&=E[Y(t)]\\&=\int_{-\infty}^{\infty}E[X(\tau)h(t-\tau)]d\tau\\&=m_x\int_{-\infty}^{\infty}h(t-\tau)d\tau\\&=m_x\int_{-\infty}^{\infty}h(\tau)d\tau\\&=m_xH(0)\end{aligned} \tag{2.5.3}$$

其中 $H(0)$是该滤波器频率响应 $H(f)$在 $f=0$ 的值。

$Y(t)$的自相关函数是

$$\begin{aligned}R_y(\tau)&=E[Y(t)Y(t+\tau)]\\&=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}E[X(\beta)X(\alpha)]h(t-\beta)h(t+\tau-\alpha)d\beta d\alpha\\&=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}R_x(\beta-\alpha)h(t-\beta)h(t+\tau-\alpha)d\beta d\alpha\end{aligned} \tag{2.5.4}$$

在频域中,输出过程 $Y(t)$的功率谱与输入过程 $X(t)$的功率谱和该线性滤波器的频率响应的关系由下式相关联:

$$S_Y(f)=S_X(f)\mid H(f)\mid^2 \tag{2.5.5}$$

通过对式(2.5.4)取傅里叶变换,很容易证明式(2.5.5)。

解说题

解说题 2.6 [过滤后的噪声]

假设有一个功率谱为$S_x(f)=1$(对所有 f)的白色随机过程 $X(t)$激励某个线性滤波器,该滤波器的冲激响应是:

$$h(t)=\begin{cases}e^{-t}, & t\leqslant 0\\0, & t<0\end{cases} \tag{2.5.6}$$

求该滤波器输出的功率谱$S_Y(f)$。

题 解

这个滤波器的频率响应很容易求出:

$$H(f)=\frac{1}{1+j2\pi f} \tag{2.5.7}$$

所以

$$S_y(f) = |H(f)|^2 = \frac{1}{1+(2\pi f)^2} \tag{2.5.8}$$

图 2.15 显示的是$S_y(f)$的图。已知$S_x(f)$和 $H(f)$，计算$S_y(f)$的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 2.6.
echo on
delta=0.01;
F_min=-2;
F_max=2;
f=F_min:delta:F_max;
Sx=ones(1,length(f));
H=1./(1+(2*pi*f).^2);
Sy=Sx.*H.^2;
```

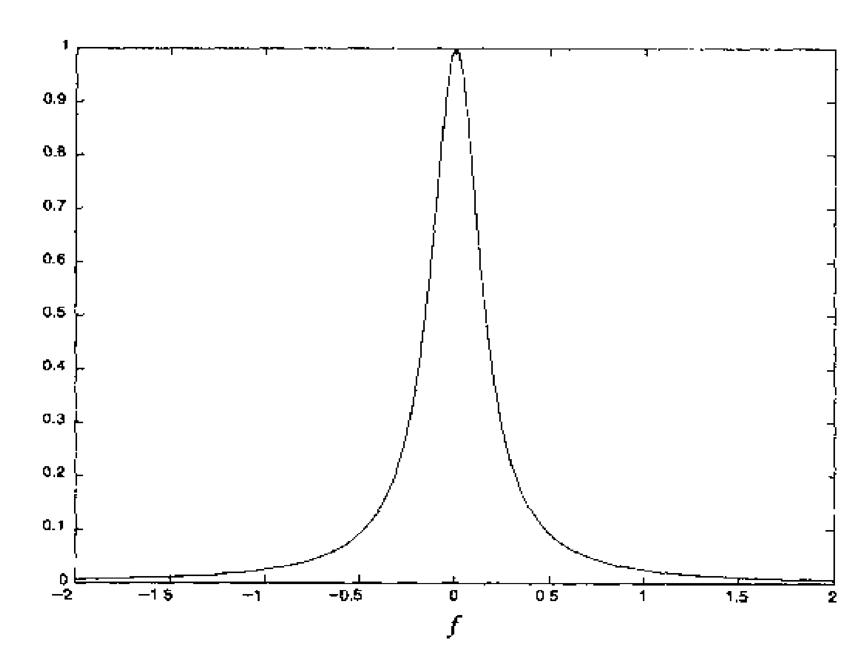


图 2.15 由式(2.5.8)给出的$S_y(f)$的图

解说题

解说题 2.7 [自相关与功率谱]

已知$S_x(f)=1$，对应于解说题 2.6 中的$S_y(f)$，计算自相关函数 $R_y(\tau)$。

题 解

在这种情况下，可以对式(2.5.8)给出的$S_y(f)$样本采用逆 FFT。图 2.16 显示的是在 $N=256$ 个频率样本和频率间隔 $\Delta f=0.1$ 时的计算结果。这个计算的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 2.7.
echo on
N=256;                                          % number of samples
deltaf=0.1;                                     % frequency separation
f=[0:deltaf:(N/2)*deltaf, -(N/2-1)*deltaf:deltaf:-deltaf];
                                                % Swap the first half.
Sy=1./(1+(2*pi*f).^2);                          % sampled spectrum
Ry=ifft(Sy);                                    % autocorrelation of Y
% Plotting command follows.
plot(fftshift(real(Ry)));
```

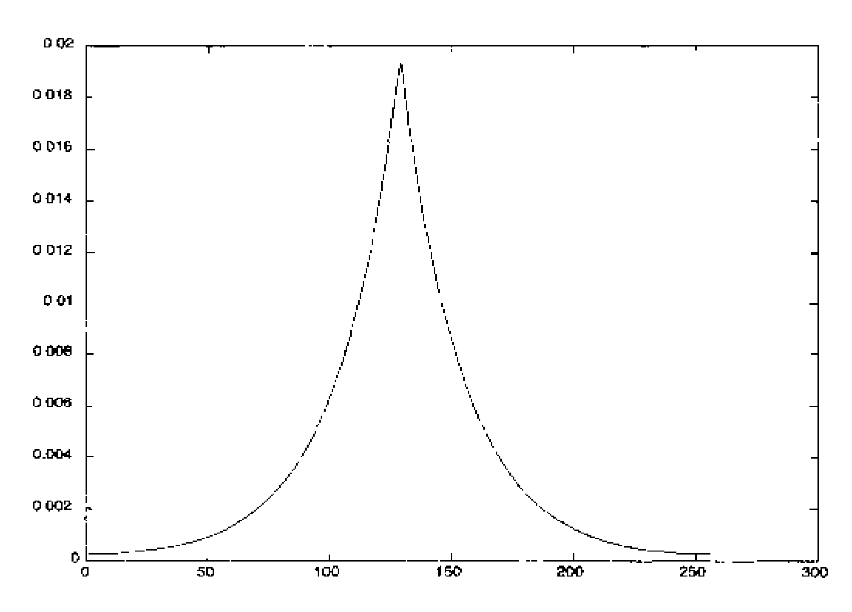


图 2.16 解说题 2.7 中的 $R_y(\tau)$的图

现在考虑等效的离散时间问题。假设一个平稳随机过程 $X(t)$被采样,其样本通过某一离散时间线性滤波器,该滤波器的脉冲响应为 $h(n)$。该线性滤波器的输出由卷积和给出:

$$Y(n) = \sum_{k=0}^{\infty} h(k)X(n-k) \tag{2.5.9}$$

其中 $X(n) \equiv X(t_n)$是输入随机过程 $X(t)$的离散时间值。$Y(n)$是离散时间滤波器的输出。输出过程的均值是:

$$\begin{aligned} m_y &= E[Y(n)] \\ &= \sum_{k=0}^{\infty} h(k)E[X(n-k)] \\ &= m_x \sum_{k=0}^{\infty} h(k) \\ &= m_x H(0) \end{aligned} \tag{2.5.10}$$

其中 $H(0)$是滤波器的频率响应 $H(f)$在 $f=0$ 的值,而 $H(f)$为:

$$H(f) = \sum_{k=0}^{\infty} h(n)\mathrm{e}^{-j2\pi fn} \tag{2.5.11}$$

输出过程的自相关函数是：

$$R_Y(m) = E[Y(n)Y(n+m)]$$
$$= \sum_{k=0}^{\infty}\sum_{l=0}^{\infty} h(k)h(l)E[X(n-k)X(n+m-l)]$$
$$= \sum_{k=0}^{\infty}\sum_{l=0}^{\infty} h(k)h(l)R_X(m-l+k) \tag{2.5.12}$$

在频域的相应表达式是：

$$\mathcal{S}_Y(f) = \mathcal{S}_X(f)\,|H(f)|^2 \tag{2.5.13}$$

其中，功率谱定义为：

$$\mathcal{S}_X(f) = \sum_{m=-\infty}^{\infty} R_X(m)e^{-j2\pi fm} \tag{2.5.14}$$

和

$$\mathcal{S}_Y(f) = \sum_{m=-\infty}^{\infty} R_Y(m)e^{-j2\pi fm} \tag{2.5.15}$$

解说题

解说题 2.8 ［过滤后的白噪声］

假设一个样本为$\{X(n)\}$的白色随机过程通过一个线性滤波器，该滤波器的脉冲响应是：

$$h(n) = \begin{cases}(0.95)^n, & n \geqslant 0\\ 0, & n < 0\end{cases}$$

求输出过程$\{Y(n)\}$的功率谱。

题　解

我们很容易看出

$$H(f) = \sum_{n=0}^{\infty} h(n)e^{-j2\pi fn}$$
$$= \sum_{n=0}^{\infty}(0.95e^{-j2\pi f})^n$$
$$= \frac{1}{1-0.95e^{-j2\pi f}} \tag{2.5.16}$$

和

$$|H(f)|^2 = \left|\frac{1}{1-0.95e^{-j2\pi f})}\right|^2$$
$$= \frac{1}{1.9025-1.9\cos(2\pi f)} \tag{2.5.17}$$

因此，输出过程的功率谱是：

$$\mathcal{S}_Y(f) = |H(f)|^2\mathcal{S}_X(f) \tag{2.5.18}$$
$$= \frac{1}{1.9025-1.9\cos(2\pi f)} \tag{2.5.19}$$

这里已假定$\mathcal{S}_X(f)$归一化到 1。$\mathcal{S}_Y(f)$如图 2.17 所示。注意，$\mathcal{S}_Y(f)$是周期的，周期为 2π。这个计算的 MATLAB 脚本如下所示。

m文件

```
% MATLAB script for Illustrative Problem 2.8.
delta_w=2*pi/100;
w=-pi:delta_w:pi;                              % one period of Sy
Sy=1./(1.9025-1.9*cos(w));
% Plotting command follows
plot(w,Sy);
```

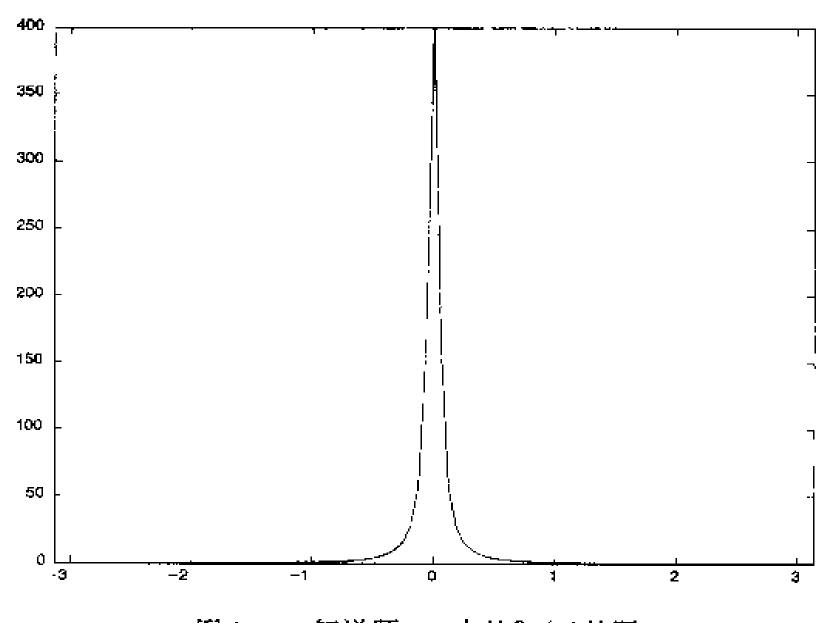


图 2.17 解说题 2.8 中的$S_y(f)$的图

输出过程$\{Y(n)\}$的自相关函数可取$S_y(f)$的逆 FFT 求出。读者将会发现,将这个自相关函数与解说题 2.3 所得出的结果进行比较,会很有意思。

2.6 低通和带通过程

和在确定性信号的情况下一样,随机信号也能由低通和带通随机过程来表征。

定义 若过程的功率谱在 $f=0$ 附近是大的,而在高频域很小(接近 0),这个随机过程就称为**低通的**。换句话说,一个低通随机过程的功率大部分集中在低频域。

定义 若过程的功率谱$S_X(f)=0, |f|>B$,则称这个低通随机过程 $X(t)$是**带限的**,参数 B 称为该随机过程的带宽。

解说题

解说题 2.9 [低通过程]

考虑用一个白噪声序列$\{X_n\}$通过一个低通滤波器产生一个低通随机过程的样本。输入序列是在$\left(-\frac{1}{2},\frac{1}{2}\right)$内的均匀分布随机变量的一个独立同分布序列,低通滤波器的脉冲响应是:

$$h(n)=\begin{cases}(0.9)^n, & n\geq 0\\ 0, & n<0\end{cases}$$

而且该滤波器由下面的输入/输出递推(差分)方程表征：

$$y_n = 0.9y_{n-1} + x_n, \quad n \geqslant 1, y_{-1} = 0$$

计算输出序列$\{y_n\}$,并按式(2.4.6)求自相关函数 $\hat{R}_x(m)$和 $\hat{R}_y(m)$。用计算 $\hat{R}_x(m)$和 $\hat{R}_y(m)$的 DFT 求功率谱$\hat{S}_x(f)$和$\hat{S}_y(f)$。

题　解

这些计算的 MATLAB 脚本如下所示。自相关函数和功率谱分别如图 2.18 和图 2.19 所示。要注意,自相关函数和功率谱都是在该随机过程 10 次实现中的平均值。

m 文件

```
% MATLAB script for Illustrative Problem 2.9.
N=1000;                                   % the maximum value of n
M=50;
Rxav=zeros(1,M+1);
Ryav=zeros(1,M+1);
Sxav=zeros(1,M+1);
Syav=zeros(1,M+1);
for i=1:10,                               % Take the ensemble average over ten realizations.
   X=rand(1,N)-(1/2);                     % Generate a uniform number sequence on (-1/2,1/2).
   Y(1)=0;
   for n=2:N, Y(n)=0.9*Y(n-1)+X(n); end;  % Note that Y(n) means Y(n-1).
   Rx=Rx_est(X,M);                        % autocorrelation of {Xn}
   Ry=Rx_est(Y,M);                        % autocorrelation of {Yn}
   Sx=fftshift(abs(fft(Rx)));             % power spectrum of {Xn}
   Sy=fftshift(abs(fft(Ry)));             % power spectrum of {Yn}
   Rxav=Rxav+Rx;
   Ryav=Ryav+Ry;
   Sxav=Sxav+Sx;
   Syav=Syav+Sy;
   echo off ;
end;
echo on ;
Rxav=Rxav/10;
Ryav=Ryav/10;
Sxav=Sxav/10;
Syav=Syav/10;
% Plotting commands follow.
```

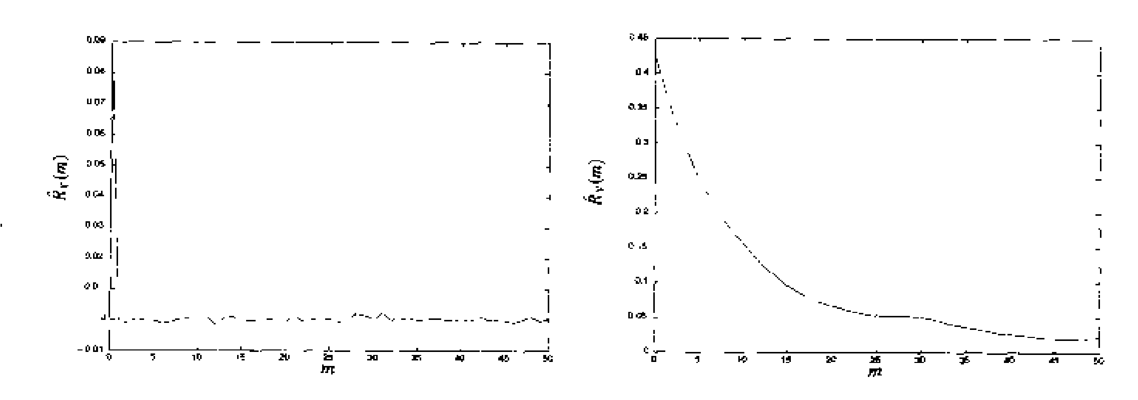


图 2.18　解说题 2.9 中的自相关函数 $\hat{R}_x(m)$和 $\hat{R}_y(m)$

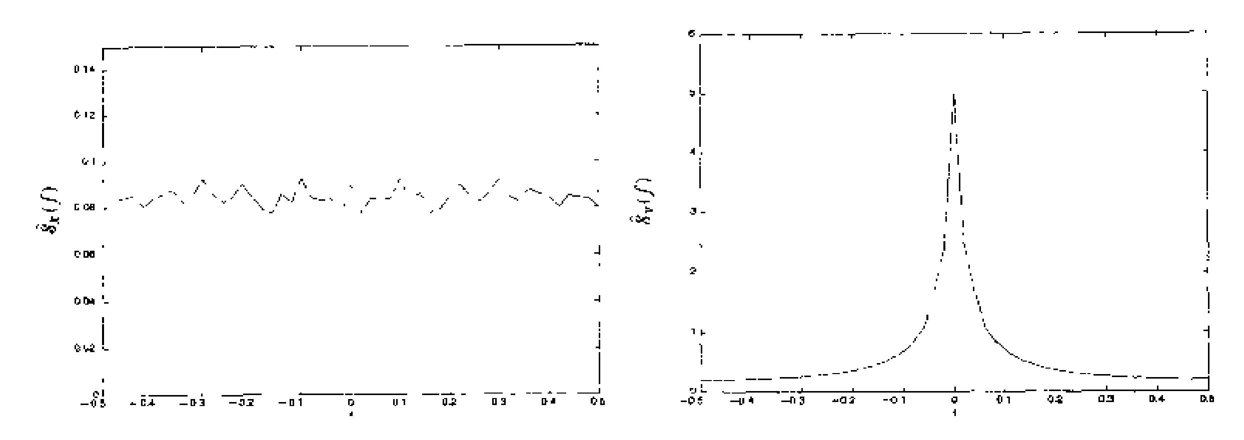


图 2.19 解说题 2.9 中的功率谱$\hat{S}_x(f)$和$\hat{S}_y(f)$

m 文件

```
function [Rx]=Rx_est(X,M)
% [Rx]=Rx_est(X,M)
%          RX_EST   estimates the autocorrelation of the sequence of random
%                   variables given in X. Only Rx(0), Rx(1), ... , Rx(M) are computed.
%                   Note that Rx(m) actually means Rx(m-1).
N=length(X);
Rx=zeros(1,M+1);
for m=1:M+1,
  for n=1:N-m+1,
    Rx(m)=Rx(m)+X(n)*X(n+m-1);
  end;
  Rx(m)=Rx(m)/(N-m+1);
end;
```

定义 若过程的功率谱在某中心频率 $\pm f_0$ 邻近的一个频带内是大的，而在该频带以外相对很小，就称这个随机过程是**带通过程**，若通带 $B \ll f_0$，则是一个**窄带**过程。

带通过程对表示调制信号是很适合的。在通信系统中，携带信息的信号通常都是一个低通随机过程，用它来调制某个载波，在一个带通窄带通信信道上传输。因此，已调制信号是一个带通随机过程。

和在确定性信号的情况下一样，带通随机过程 $X(t)$也可以表示成

$$X(t) = X_c(t)\cos(2\pi f_0 t) - X_s(t)\sin(2\pi f_0 t) \tag{2.6.1}$$

其中 $X_c(t)$和 $X_s(t)$称为 $X(t)$的同相和正交分量。随机过程 $X_c(t)$和 $X_s(t)$都是低通过程。下面的定理(未给出证明)给出了 $X(t)$，$X_c(t)$和 $X_s(t)$之间的重要关系。

定理 若 $X(t)$是一个零均值的平稳随机过程，那么过程 $X_c(t)$和 $X_s(t)$也是零均值的联合平稳过程。

事实上，很容易证明(见参考文献[1])，$X_c(t)$和 $X_s(t)$的自相关函数是相同的，并且可以表示为：

$$R_c(\tau) = R_s(\tau) = R_x(\tau)\cos(2\pi f_0\tau) + \hat{R}_x(\tau)\sin(2\pi f_0\tau) \tag{2.6.2}$$

其中 $R_x(\tau)$是带通过程 $X(t)$的自相关函数，而 $\hat{R}_x(\tau)$是 $R_x(\tau)$的希尔伯特变换，定义为：

$$\hat{R}_s(\tau) = \frac{1}{\pi}\int_{-\infty}^{\infty} \frac{R(t)}{\tau - t}\mathrm{d}t \tag{2.6.3}$$

另外，$X_c(t)$和 $X_s(t)$的互相关函数表示为：

$$R_{cs}(\tau) = R_\tau(\tau)\sin(2\pi f_0\tau) - \hat{R}_\tau(\tau)\cos(2\pi f_0\tau) \tag{2.6.4}$$

最后，使用自相关函数 $R_c(\tau)$和互相关函数 $R_{cs}(\tau)$，可以将该带通过程 $X(t)$的自相关函数表示成：

$$R_x(\tau) = R_c(\tau)\cos(2\pi f_0\tau) - R_{cs}(\tau)\sin(2\pi f_0\tau) \tag{2.6.5}$$

解说题

解说题 2.10 ［带通随机过程样本的产生］

按这里给出的方法产生一个高斯带通随机过程的样本。首先产生两个统计独立的高斯随机过程 $X_c(t)$和 $X_s(t)$，然后分别用它们来调制两个互为正交的载波 $\cos(2\pi f_0 t)$和 $\sin(2\pi f_0 t)$，如图 2.20 所示。

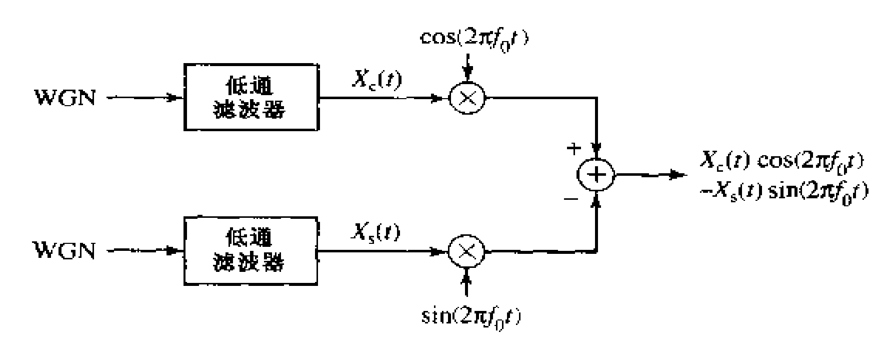


图 2.20 带通随机过程的产生

题 解

在一台计算机上，将两个独立的高斯白噪声过程经由两个完全一样的低通滤波器过滤，产生低通过程 $X_c(t)$和 $X_s(t)$的样本。由此，可以得到样本 $X_c(n)$和 $X_s(n)$，这就相应于 $X_c(t)$和 $X_s(t)$的采样值。然后，用 $X_c(n)$调制已采样的载波 $\cos(2\pi f_0 nt)$，用 $X_s(n)$调制正交载波 $\sin(2\pi f_0 nT)$，其中 T 是合适的采样间隔。

这些计算的 MATLAB 脚本如下所示。为了说明起见，已选取的低通滤波器的传递函数为

$$H(z) = \frac{1}{1 - 0.9z^{-1}}$$

并且，取 $T = 1$ 和 $f_0 = 1000/\pi$。这个带通过程所得到的功率谱如图 2.21 所示。

m 文件

```
% MATLAB script for Illustrative Problem 2.10.
N=1000;                                  % number of samples
for i=1:2:N,
  [X1(i) X1(i+1)]=gngauss;
  [X2(i) X2(i+1)]=gngauss;
  echo off ;
end;                                     % standard Gaussian input noise processes
echo on ;
```

```
A=[1 -0.9];                              % lowpass filter parameters
B=1;
Xc=filter(B,A,X1);
Xs=filter(B,A,X2);
fc=1000/pi;                              % carrier frequency
for i=1:N,
    band_pass_process(i)=Xc(i)*cos(2*pi*fc*i)-Xs(i)*sin(2*pi*fc*i);
    echo off ;
end;                                     % T=1 is assumed.
echo on;
% Determine the autocorrelation and the spectrum of the bandpass process.
M=50;
bpp_autocorr=Rx_est(band_pass_process,M);
bpp_spectrum=fftshift(abs(fft(bpp_autocorr)));
% Plotting commands follow.
```

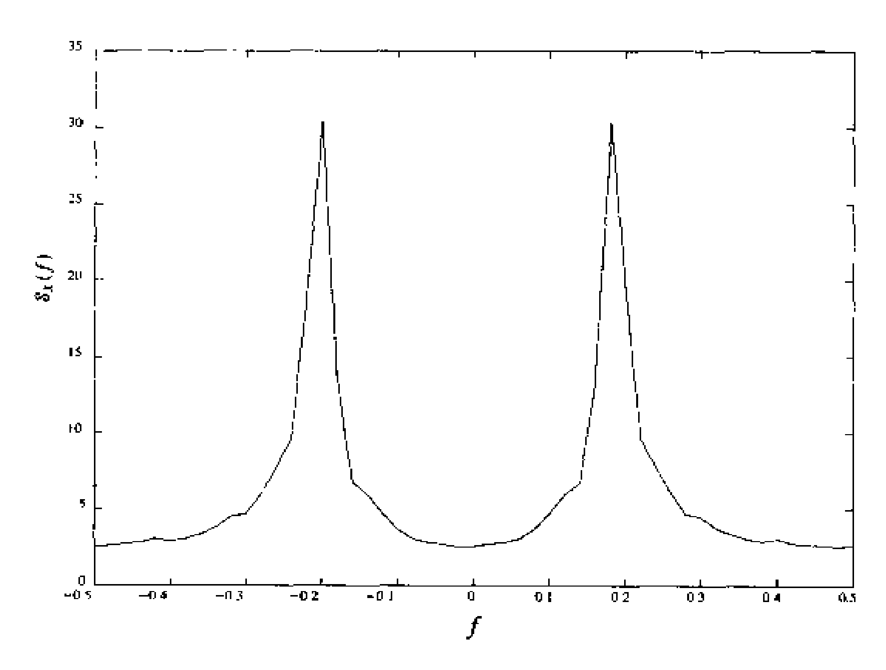


图 2.21 解说题 2.10 中的带通过程的功率谱

2.7 数字通信系统的 Monte Carlo 仿真

在实际中,Monte Carlo 计算机仿真通常用于在有噪声和干扰的情况下对一个数字通信系统的性能做出估计。用于评价数字通信系统性能的最常用指标是差错概率。

为了阐明估计关于一个随机变量的某种概率的过程,现考虑随机变量

$$Y = m + G \tag{2.7.1}$$

其中,m 是某个常数,G 是一个零均值、方差 $\sigma^2 = 1$ 的高斯随机变量。显然,Y 是一个均值为 m 和方差 $\sigma^2 = 1$ 的高斯随机变量。

设想对某个给定的 m 值,利用一台计算机完成一系列试验(Monte Carlo 仿真)。我们想要估计出 $Y < 0$ 的概率,即

$$P(m) = P(Y < 0 \mid m) \tag{2.7.2}$$

具体地说，$P(m)$估计可以这样得到。正如 2.2 节所述，利用一台计算机产生一个统计独立的、同分布的、零均值的单位方差高斯随机变量 $G_i, i=1,2,\cdots,N$；然后，将一个常数值 m 加在每个 G_i 上，得到随机变量序列：

$$Y_i = m + G_i, \quad i = 1,2,\cdots,N \tag{2.7.3}$$

假定 m 是某个正值，那么 Y_i 的概率密度函数如图 2.22 所示。

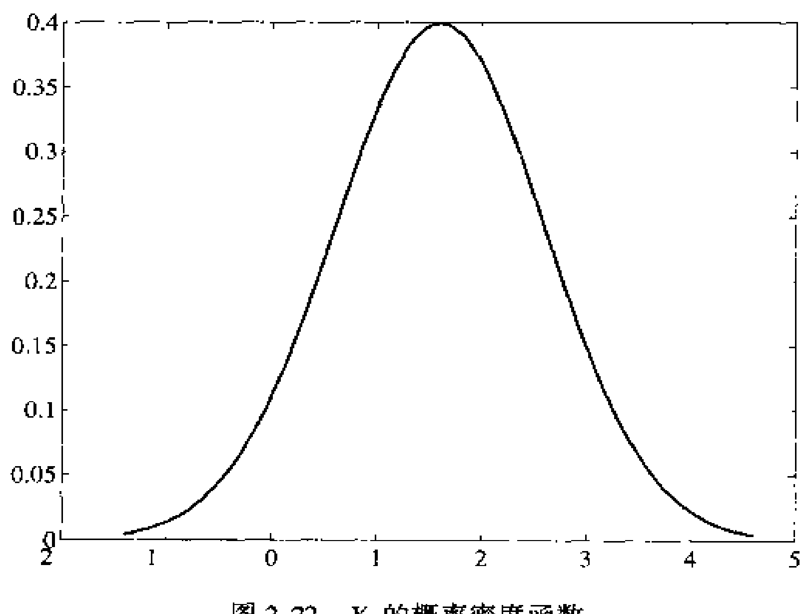


图 2.22　Y_i 的概率密度函数

从这些由计算机产生的随机变量中，我们希望估计出具有均值 $m(m>0)$和单位方差的一个高斯随机变量小于零的概率。很显然，这个估计的值就等于在图 2.22 中的概率密度函数后部下面的面积。因此，可以测试每个 Y_i，看它是否有 $Y_i<0$，这样就能够定义一个新的随机变量 X_i，如下所示：

$$X_i = \begin{cases} 0, & 若\ Y_i \geqslant 0 \\ 1, & 若\ Y_i < 0 \end{cases} \tag{2.7.4}$$

这样，概率 $P(Y<0|m)$的估计是

$$\hat{P}(m) = \frac{1}{N}\sum_{i=1}^{N} X_i \tag{2.7.5}$$

换句话说，$P(m)$的估计就是随机变量 $Y_i, i=1,2,\cdots,N$ 小于零的个数除以总的随机变量数 N。

由于估计 $\hat{P}(m)$是随机变量 $X_i, i=1,2,\cdots,N$ 的函数，所以它也是一个随机变量。为了确定这个估计逼近真正的 $P(m)$有多好，可以计算出这个估计的均值和方差。首先，$\hat{P}(m)$的均值是：

$$E[\hat{P}(m)] = \frac{1}{N}\sum_{i=1}^{N} E(X_i)$$

但是

$$E(X_i) = 0 \cdot P(Y_i \geqslant 0) + 1 \cdot P(Y_i < 0)$$

因此有：

$$E[\hat{P}(m)] = \frac{1}{N}\sum_{i=1}^{N} P(Y_i < 0)$$

$$= \frac{1}{N}\sum_{i=1}^{N} P(m) = P(m) \tag{2.7.6}$$

这说明估计 $\hat{P}(m)$的均值等于真正的 $P(m)$的均值。满足这个性质的估计称为**无偏估计**。

下面计算估计 $\hat{P}(m)$的方差。$\hat{P}(m)$的方差定义为：

$$\sigma^2_{P(m)} = E\left[\hat{P}(m) - E[\hat{P}(m)]\right]^2 \tag{2.7.7}$$

$$= E[\hat{P}^2(m)] - P^2(m) \tag{2.7.8}$$

但是，

$$E[\hat{P}^2(m)] = E\left[\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} X_i X_j\right]$$

$$= \frac{1}{N^2}\sum_{i=1}^{N} E[X_i^2] + \frac{1}{N^2}\sum_{\substack{i=1 \\ i\neq j}}^{N}\sum_{j=1}^{N} E(X_i X_j) \tag{2.7.9}$$

然后有

$$E[X_i^2] = 0 \cdot P(Y_i \geqslant 0) + 1P(Y_i < 0)$$

$$= P(m) \tag{2.7.10}$$

和

$$E(X_i X_j) = E(X_i)E(X_j) = P^2(m) \tag{2.7.11}$$

将式(2.7.9)和式(2.7.10)代入式(2.7.8)，得到的结果为：

$$E[\hat{P}^2(m)] = \frac{1}{N}P(m) + \frac{N(N-1)}{N^2}P^2(m)$$

$$= \frac{1}{N}P(m)[1 - P(m)] + P^2(m) \tag{2.7.12}$$

最后将式(2.7.12)的结果代入式(2.7.7)，求得估计的方差为：

$$\sigma^2_{P(m)} = \frac{1}{N}P(m)[1 - P(m)] \tag{2.7.13}$$

一般来说，当基于 Monte Carlo 方法对 $P(m)$求估计时，我们希望预测出估计的标准偏差$\sigma_{P(m)}$与 $P(m)$相比更小。例如，设想正在对某个小的概率(如 $P(m) = 10^{-3}$)进行估计，问需要取多大的样本数才能保证标准偏差 $\sigma_{P(m)}$ 与 $P(m)$相比更小，也就是说，

$$\sigma_{P(m)} = \left[\frac{1}{N}P(m)[1 - P(m)]\right]^{1/2} \ll P(m)$$

等效为 $\sigma^2_{P(m)} \ll P^2(m)$。由于 $P(m) \ll 1$，可以得到：

$$\frac{1}{N}P(m) \ll P^2(m)$$

据此有

$$N \gg \frac{1}{P(m)} \tag{2.7.14}$$

例如，若 $P(m) = 10^{-3}$，那么 $N \gg 1000$。如果选取样本大小为 $N = 10\,000$，平均会有 $Y_i, i = 1, 2, \cdots, 10\,000$ 的 10 个值是小于零的，一般我们就认为，为了得到一个可靠的 $P(m)$估计，这个样本大小是最小的值。所以，作为经验公式，当 $P(m) \ll 1$ 时，样本大小应满足条件：

$$N > \frac{10}{P(m)} \tag{2.7.15}$$

在加性噪声和其他加性干扰存在的情况下，对数字通信系统的差错概率进行估计时，这个条件通常就足够了。

解说题

解说题 2.11 [在 Monte Carlo 仿真中的样本大小]

在数字通信系统中，具有电压电平 $m(m>0)$ 的接收信号受到零均值、单位方差和加性高斯噪声的污损。当 $Y_i = m + G_i$，$m=3$ 和 $m=5$ 时，基于式(2.7.15)给出的经验公式，为了确定概率 $P(Y<0|m)$，求所需要的最小样本数。

题 解

通过 Monte Carlo 仿真，要被估计的真正概率是：

$$m = 3:\quad P(3) = P(Y < 0 \mid m = 3) = 1.35 \times 10^{-3}$$

$$m = 5:\quad P(5) = P(Y < 0 \mid m = 5) = 2.87 \times 10^{-7}$$

根据式(2.7.15)给出的经验公式，估计 $P(3)$ 和 $P(5)$ 所需要的最小样本数为：

$$m = 3:\quad N = \frac{10}{P(3)} = 7047$$

$$m = 5:\quad N = \frac{10}{P(5)} = 3.48 \times 10^{7}$$

2.8 习题

2.1 利用 MATLAB 函数 rand(1,N)在区间[0,1]上产生 1000 个均匀随机数的集合，画出这个序列的直方图和概率分布函数。直方图可以这样得到：用覆盖[0,1]范围的 10 个等宽子区间来量化这个区间，并在每个子区间内计数。

2.2 利用 MATLAB 函数 rand(1,N)在区间$\left[-\frac{1}{2},\frac{1}{2}\right]$上产生 1000 个均匀随机数的集合，画出这个序列的直方图和概率分布函数。

2.3 利用 MATLAB 函数 rand(1,N)在区间[−2,2]上产生 1000 个均匀随机数的集合，画出这个序列的直方图和概率分布函数。

2.4 产生具有如下线性概率密度函数：

$$f(x) = \begin{cases} \dfrac{x}{2}, & 0 \leqslant x \leqslant 2 \\ 0, & \text{其余 } x \end{cases}$$

的 1000 个随机数，画出直方图和概率分布函数。

2.5 利用 2.2 节叙述的方法，产生具有零均值和单位方差的 1000 个高斯随机数的集合，画出这个序列的直方图和概率分布函数。在画直方图时，随机数的范围可以划分成宽度为 $\sigma^2/5$ 的子区间，从包括 $-\sigma^2/10 < x < \sigma^2/10$ 范围的第一个区间开始，这里的 σ^2 是方差。

2.6 利用 MATLAB 函数 rand(1,N)产生具有零均值和单位方差的 1000 个高斯随机数的集合，画出这个序列的直方图和概率分布函数。将这个结果与习题 2.5 中的结果进行比较。

2.7 中心极限定理讲的是，若随机变量 X_i，$1 \leqslant i \leqslant n$ 是独立同分布的，具有有限均值和方差，并且 n 很大，那么它们的平均(即 $Y = \frac{1}{n}\sum_{i=1}^{n} X_i$)就大致上具有一个高斯分布。这个定理说明了为什么在电路中产生的热噪声具有高斯分布。本题要用 MATLAB 验证这个定理。

a. 利用 MALTAB 产生一个长度为 1 000 000 的向量 $\boldsymbol{x}$，其分量都是在 0 和 1 之间均匀分布的随机变量。在中心极限定理中，这个向量的分量是 X_i(利用 MATLAB 命令 rand 产生这个序列)。

b. 求每 100 个连续的 $\boldsymbol{x}$ 的分量的平均，并产生长度为 10 000 的序列 $\boldsymbol{y}$。这样，Y_1 是 X_1 到 X_{100}的平均；Y_2 是 X_{101} 到 X_{200}的平均，依次类推。

c. 利用 hist 命令画出序列 $\boldsymbol{y}$ 的直方图，用 40 个二进制位来产生此图，结果要包括程序清单和此图。注意，这个直方图是非常接近于高斯分布的。

2.8 产生 1000 对高斯随机数(x_1, x_2)，它们具有均值向量

$$\boldsymbol{m} = E[x_1 \ x_2] = \left[\frac{1}{2} \ \frac{1}{2}\right]$$

和协方差矩阵

$$\boldsymbol{C} = \begin{bmatrix} 1 & \frac{1}{2} \\ \frac{1}{2} & 1 \end{bmatrix}$$

a. 求样本(x_{1i}, x_{2i})，$i = 1, 2, \cdots, 1000$ 的均值，均值定义为：

$$\hat{m}_1 = \frac{1}{1000}\sum_{i=1}^{1000} x_{1i}$$

$$\hat{m}_2 = \frac{1}{1000}\sum_{i=1}^{1000} x_{2i}$$

并且，求它们的方差

$$\hat{\sigma}_1^2 = \frac{1}{1000}\sum_{i=1}^{1000}(x_{1i} - \hat{m}_1)^2$$

$$\hat{\sigma}_2^2 = \frac{1}{1000}\sum_{i=1}^{1000}(x_{2i} - \hat{m}_2)^2$$

和协方差

$$\hat{c}_{ij} = \frac{1}{1000}\sum_{i=1}^{1000}(x_{1i} - \hat{m}_1)(x_{2i} - \hat{m}_2)$$

b. 将从样本求得的这些值与理论值进行比较。

2.9 产生高斯 - 马尔可夫过程的 1000 个样本序列，该过程由下面的递推关系描述：

$$X_n = \rho X_{n-1} + W_n, \quad n = 1, 2, \cdots, 1000$$

其中 $X_0 = 0$，$\rho = 0.9$，而$\{W_n\}$是一个零均值、单位方差、独立同分布的高斯随机变量。

2.10 用一个独立同分布的零均值、单元方差的高斯随机变量重做解说题 2.4。

2.11 当一个带限随机过程的功率谱如下所示时：

$$S_x(f) = \begin{cases} 1 - \frac{|f|}{B}, & |f| \leqslant B \\ 0, & \text{其余 } f \end{cases}$$

重做解说题 2.5。

2.12 当线性滤波器的冲激响应如下所示时：

$$h(t)=\begin{cases}e^{-3t}, & t\geqslant 0\\ 0, & t<0\end{cases}$$

重做解说题 2.6。

2.13 用数值法求习题 2.12 中的线性滤波器输出端的随机过程的自相关函数。

2.14 当 $h(n)$如下所示时：

$$h(n)=\begin{cases}(0.8)^n, & n\geqslant 0\\ 0, & n<0\end{cases}$$

重做解说题 2.8。

2.15 产生一个在$\left[-\frac{1}{2},\frac{1}{2}\right]$内，$N=1000$ 的均匀分布随机数的独立同分布序列$\{x_n\}$。这个序列通过一个线性滤波器，其脉冲响应为：

$$h(n)=\begin{cases}(0.95)^n, & n\geqslant 0\\ 0, & n<0\end{cases}$$

作为输入的函数，描述这个滤波器的输出的递推公式是：

$$y_n=0.95y_{n-1}+x_n \quad n\geqslant 0, y_{-1}=0$$

利用式(2.4.6)和式(2.4.7)给出的关系，计算这个序列$\{x_n\}$和$\{y_n\}$的自相关函数$\hat{R}_x(m)$和 $\hat{R}_y(m)$，及其相应的功率谱$\hat{S}_x(f)$和$\hat{S}_y(f)$。将$\hat{S}_y(f)$的结果与解说题 2.8 中得到的结果进行比较。

2.16 产生两个在$\left[-\frac{1}{2},\frac{1}{2}\right]$内，$N=1000$ 的均匀分布随机数的独立同分布序列$\{w_{cn}\}$和$\{w_{sn}\}$，每个序列都通过一个线性滤波器，其脉冲响应为：

$$h(n)=\begin{cases}\left(\frac{1}{2}\right)^n, & n\geqslant 0\\ 0, & n<0\end{cases}$$

该滤波器的输入/输出特性由下面的递推关系给出：

$$x_n=\frac{1}{2}x_{n-1}+w_n, \quad n\geqslant 1, x_0=0$$

由此得到两个序列$\{x_{cn}\}$和$\{x_{sn}\}$，输出序列$\{x_{cn}\}$调制载波 $\cos(\pi/2)n$，输出序列$\{x_{sn}\}$调制正交载波 $\sin(\pi/2)n$。按式(2.6.1)将这两个已调分量组合起来就构成了带通信号。

分别对序列$\{x_{cn}\}$和$\{x_{sn}\}$计算并画出自相关函数分量 $\hat{R}_c(m)$和 $\hat{R}_s(m)$，$|m|\leqslant 10$。计算带通信号的自相关函数 $\hat{R}_x(m)$，$|m|\leqslant 10$。用 DFT(或 FFT 算法)计算功率谱$\hat{S}_c(f)$，$\hat{S}_s(f)$和$\hat{S}_x(f)$。画出这些功率谱，并对这些结果进行讨论。

2.17 使用 Monte Carlo 仿真，当 $Y=m+G, m=3$，并且 G 是一个零均值和单位方差的高斯随机变量时，估计概率 $P(m)=p(Y<0|m)$，将这个仿真试验重复 5 次，并在 5 次试验中采用不同的高斯随机数的集合观察 $P(m)$的估计。讨论在 5 次估计中所观察到的变化，以及估计与真值 $P(3)$的匹配程度。

第3章 模拟调制

3.1 概述

第3章要讨论各种模拟调制解调系统的性能(包括有加性噪声和无加性噪声存在时的情况),要讨论的系统包括幅度调制(AM)方法(例如DSB-AM、SSB-AM和常规AM)和角调制方法(例如频率与相位调制)。

每种模拟调制系统都用下面五个方面的基本性质来表征:

1. 已调信号的时域表示;
2. 已调信号的频域表示;
3. 已调信号的带宽;
4. 已调信号的功率含量;
5. 解调后的信噪比(SNR)。

很显然,这些性质互相之间不是独立的。通过傅里叶变换关系,可以反映出信号的时域和频域表示之间存在着很紧密的关系;另外,信号的带宽也是用它的频域特性来定义的。

由于幅度调制和角调制方法存在着本质差异,所以在不同的小节里讨论了这两种方法。我们首先从最简单的调制方法——幅度调制开始讨论。

3.2 幅度调制

幅度调制(Amplitude Modulation,简称AM)往往为**线性调制**,它是调制方法中的一类,其中一个正弦载波的幅度作为调制信号的函数而改变。这一类调制方法由DSB-AM(双边带幅度调制)、常规幅度调制、SSB-AM(单边带幅度调制)和VSB-AM(残留边带幅度调制)等组成。调制信号和已调载波幅度之间的依赖关系可以很简单,如DSB-AM,也可以很复杂,如SSB-AM或VSB-AM。与角调制方法相比,幅度调制的通常特点是带宽要求相对较低,并且功率效率不高。AM系统的带宽要求在W和$2W$之间变化,这里的W是消息信号的带宽。对于SSB-AM,带宽是W;对于DSB-AM和常规AM,带宽都是$2W$;对于VSB-AM,带宽则位于W和$2W$之间。这些系统广泛用于广播(AM无线电广播和TV视频广播)、点对点通信(SSB)和多路复用(如很多电话信道在微波链路上传输)等系统中。

3.2.1 DSB-AM

在DSB-AM中,已调信号的幅度正比于消息信号。这意味着,已调信号的时域表示由下式给出:

$$u(t) = A_c m(t)\cos(2\pi f_c t) \tag{3.2.1}$$

其中

$$c(t)=A_c\cos(2\pi f_c t) \tag{3.2.2}$$

是载波，而 $m(t)$是消息信号。取 $u(t)$的傅里叶变换，可以得到 DSB-AM 信号的频域表示为：

$$U(f)=\frac{A_c}{2}M(f-f_c)+\frac{A_c}{2}M(f+f_c) \tag{3.2.3}$$

其中 $M(f)$是 $m(t)$的傅里叶变换。很明显，这种调制形式将消息信号的频谱做了 $\pm f_c$ 的频移，并在幅度上乘以 $A_c/2$。传输带宽 B_T 是消息信号带宽的两倍，即：

$$B_T=2W \tag{3.2.4}$$

图 3.1 显示的是一个典型的消息信号频谱及其相应的 DSB-AM 已调信号的频谱。

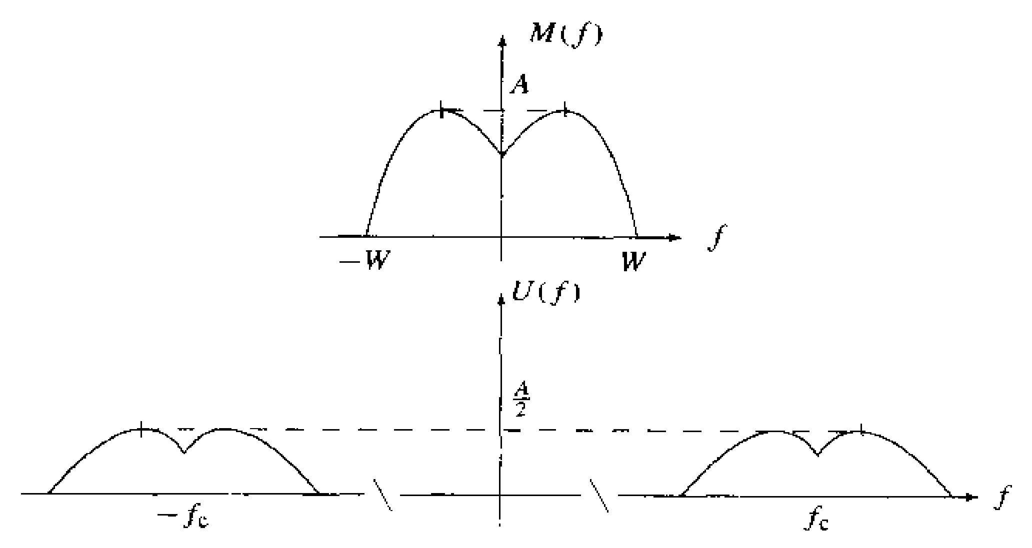


图 3.1　消息信号和 DSB-AM 已调信号的频谱

已调信号的功率由下式给出：

$$\begin{aligned}P_u&=\lim_{T\to\infty}\frac{1}{T}\int_{-T/2}^{T/2}u^2(t)\mathrm{d}t\\&=\lim_{T\to\infty}\frac{1}{T}\int_{-T/2}^{T/2}A_c^2m^2(t)\cos^2(2\pi f_c t)\mathrm{d}t\\&=\lim_{T\to\infty}\frac{1}{T}\int_{-T/2}^{T/2}A_c^2m^2(t)\frac{1+\cos(4\pi f_c t)}{2}\mathrm{d}t\\&=A_c^2\left\{\lim_{T\to\infty}\frac{1}{T}\int_{-T/2}^{T/2}\frac{m^2(t)}{2}\mathrm{d}t+\lim_{T\to\infty}\frac{1}{T}\int_{-T/2}^{T/2}m^2(t)\frac{\cos(4\pi f_c t)}{2}\mathrm{d}t\right\}\\&=A_c^2\lim_{T\to\infty}\frac{1}{T}\int_{-T/2}^{T/2}\frac{m^2(t)}{2}\mathrm{d}t &(3.2.5)\\&=\frac{A_c^2}{2}P_m &(3.2.6)\end{aligned}$$

其中 P_m 是消息信号的功率。式(3.2.5)是根据这一点直接得出的：$m(t)$是一个低通信号，其频率分量远远小于 $2f_c$，即 $\cos(4\pi f_c t)$的频率分量，因此下面的积分

$$\int_{-T/2}^{T/2}m^2(t)\frac{\cos(4\pi f_c t)}{2}\mathrm{d}t \tag{3.2.7}$$

随 $T\to\infty$ 而趋近于零。最后，DSB-AM 系统的 SNR 等于基带的 SNR，即：

$$\left(\frac{S}{N}\right)_{o}=\frac{P_{R}}{N_{0}W} \tag{3.2.8}$$

其中 P_R 是接收到的功率(在接收端已调信号的功率),$N_0/2$ 是噪声功率谱密度(假定为白噪声),W 是消息信号的带宽。

解说题

解说题 3.1 [DSB-AM 调制]

消息信号 $m(t)$定义为:

$$m(t)=\begin{cases}1, & 0\leqslant t\leqslant \frac{t_0}{3}\\ -2, & \frac{t_0}{3}<t\leqslant \frac{2t_0}{3}\\ 0, & \text{其余 } t\end{cases}$$

该消息以 DSB-AM 方式调制载波 $c(t)=\cos(2\pi f_c t)$,所得已调信号记为 $u(t)$。假设 $t_0=0.15$ s,$f_c=250$ Hz。

1. 求 $u(t)$的表达式。
2. 导出 $m(t)$和 $u(t)$的频谱。
3. 假定消息信号是周期的,周期 $T_0=t_0$,求在已调信号中的功率。
4. 在本题第 3 步,若噪声加到已调信号上,所得 SNR 是 10 dB,求噪声功率。

题 解

1. $m(t)$可以写成:

$$m(t)=\Pi\left(\frac{t-t_0/6}{t_0/3}\right)-2\Pi\left(\frac{t-t_0/2}{t_0/3}\right)$$

因此有:

$$u(t)=\left[\Pi\left(\frac{t-0.025}{0.05}\right)-2\Pi\left(\frac{t-0.075}{0.05}\right)\right]\cos(500\pi t) \tag{3.2.9}$$

2. 利用标准傅里叶变换关系$\mathcal{F}[\Pi(t)]=\text{sinc}(t)$,再结合傅里叶变换的时移和尺度变换定理,可得:

$$\begin{aligned}\mathcal{F}[m(t)]&=\frac{t_0}{3}e^{-j\pi f t_0/3}\text{sinc}\left(\frac{t_0 f}{3}\right)-2\frac{t_0}{3}e^{-j\pi f t_0}\text{sinc}\left(\frac{t_0 f}{3}\right)\\&=\frac{t_0}{3}e^{-j\pi f t_0/3}\text{sinc}\left(\frac{t_0 f}{3}\right)(1-2e^{-j2\pi t_0 f/3})\end{aligned} \tag{3.2.10}$$

将 $t_0=0.15$ s 代入后,得到:

$$\mathcal{F}[m(t)]=0.05e^{-0.05j\pi f}\text{sinc}(0.05f)(1-2e^{-0.1j\pi f}) \tag{3.2.11}$$

对于已调信号 $u(t)$,则有:

$$\begin{aligned}U(f)=&0.025e^{-0.05j\pi(f-f_c)}\text{sinc}[0.05(f-f_c)](1-2e^{-0.1j\pi(f-f_c)})\\&+0.025e^{-0.05j\pi(f+f_c)}\text{sinc}[0.05(f+f_c)](1-2e^{-0.1j\pi(f+f_c)})\end{aligned}$$

消息信号和已调信号的幅度谱如图 3.2 所示。

3. 在已调信号中的功率为:

$$P_u = \frac{A_c^2}{2}P_m = \frac{1}{2}P_m$$

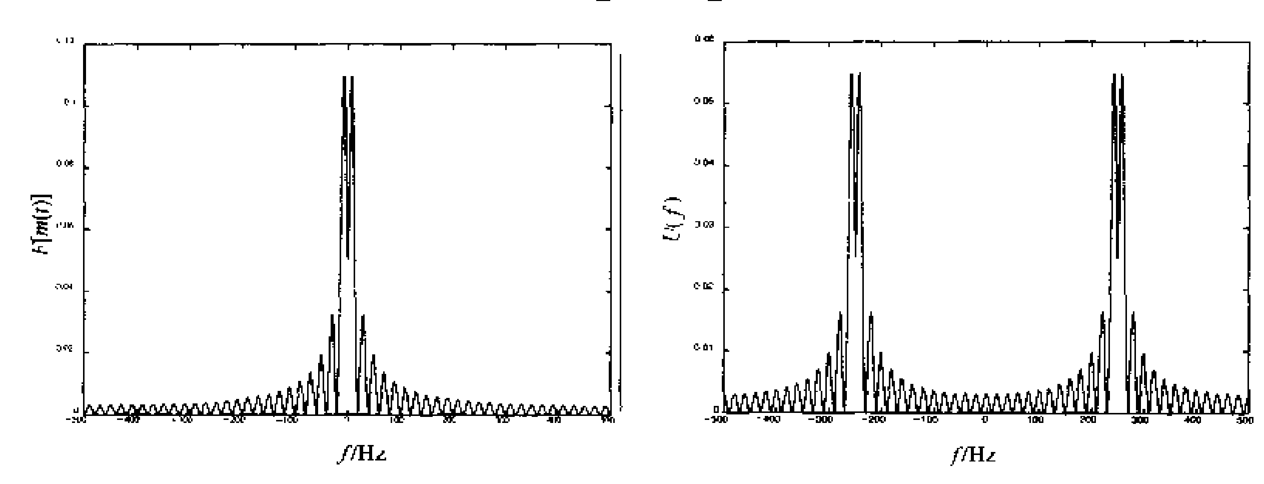


图 3.2　解说题 3.1 中的消息和已调信号的幅度谱

其中 P_m 是消息信号的功率，

$$P_m = \frac{1}{t_0}\int_0^{2t_0/3} m^2(t)\mathrm{d}t = \frac{1}{t_0}\left(\frac{t_0}{3}+\frac{4t_0}{3}\right) = \frac{5}{3} = 1.666$$

并且有

$$P_u = \frac{1.666}{2} = 0.833$$

4. 这里

$$10\log_{10}\left(\frac{P_R}{P_n}\right) = 10$$

或者 $P_R = P_u = 10P_n$，即可得到 $P_n = P_u/10 = 0.0833$。

本题的 MATLAB 脚本如下。

m 文件

```
% MATLAB script for Illustrative Problem 3.1.
% Demonstration script for DSB-AM. The message signal is
% +1 for 0 < t < t0/3, -2 for t0/3 < t < 2t0/3, and zero otherwise.
echo on
t0=.15;                              % signal duration
ts=0.001;                            % sampling interval
fc=250;                              % carrier frequency
snr=20;                              % SNR in dB (logarithmic)
fs=1/ts;                             % sampling frequency
df=0.3;                              % desired freq. resolution
t=[0:ts:t0];                         % time vector
snr_lin=10^(snr/10);                 % linear SNR
% message signal
m=[ones(1,t0/(3*ts)),-2*ones(1,t0/(3*ts)),zeros(1,t0/(3*ts)+1)];
c=cos(2*pi*fc.*t);                   % carrier signal
u=m.*c;                              % modulated signal
[M,m,df1]=fftseq(m,ts,df);           % Fourier transform
M=M/fs;                              % scaling
[U,u,df1]=fftseq(u,ts,df);           % Fourier transform
U=U/fs;                              % scaling
```

```
[C,c,df1]=fftseq(c,ts,df);                    % Fourier transform
f=[0:df1:df1*(length(m)-1)]-fs/2;             % freq  vector
signal_power=spower(u(1:length(t)));          % power in modulated signal
noise_power=signal_power/snr_lin;             % Compute noise power.
noise_std=sqrt(noise_power);                  % Compute noise standard deviation.
noise=noise_std*randn(1,length(u));           % Generate noise.
r=u+noise;                                    % Add noise to the modulated signal.
[R,r,df1]=fftseq(r,ts,df);                    % spectrum of the signal+noise
R=R/fs;                                       % scaling
pause  % Press a key to show the modulated signal power.
signal_power
pause  % Press any key to see a plot of the message.
clf
subplot(2,2,1)
plot(t,m(1:length(t)))
xlabel('Time')
title('The message signal')
pause  % Press any key to see a plot of the carrier
subplot(2,2,2)
plot(t,c(1:length(t)))
xlabel('Time')
title('The carrier')
pause  % Press any key to see a plot of the modulated signal.
subplot(2,2,3)
plot(t,u(1:length(t)))
xlabel('Time')
title('The modulated signal')
pause    % Press any key to see plots of the magnitude of the message and the
         % modulated signal in the frequency domain.
subplot(2,1,1)
plot(f,abs(fftshift(M)))
xlabel('Frequency')
title('Spectrum of the message signal')
subplot(2,1,2)
plot(f,abs(fftshift(U)))
title('Spectrum of the modulated signal')
xlabel('Frequency')
pause  % Press a key to see a noise sample.
subplot(2,1,1)
plot(t,noise(1:length(t)))
title('Noise sample')
xlabel('Time')
pause  % Press a key to see the modulated signal and noise.
subplot(2,1,2)
plot(t,r(1:length(t)))
title('Signal and noise')
xlabel('Time')
pause  % Press a key to see the modulated signal and noise in freq. domain.
subplot(2,1,1)
plot(f,abs(fftshift(U)))
title('Signal spectrum')
xlabel('Frequency')
subplot(2,1,2)
plot(f,abs(fftshift(R)))
title('Signal and noise spectrum')
xlabel('Frequency')
```

解说题

解说题 3.2　[近似于带限信号的 DSB 调制]

消息信号 $m(t)$为:

$$m(t)=\begin{cases}\text{sinc}(100t), & |t|\leqslant t_0\\ 0, & \text{其余 } t\end{cases}$$

其中 $t_0=0.1$ s。用这个消息来调制载波 $c(t)=\cos(2\pi f_c t)$, $f_c=250$ Hz。

1. 求已调信号 $u(t)$。
2. 导出 $m(t)$和 $u(t)$的频谱。
3. 假定消息信号是周期的，周期 $T_0=0.2$ s，求在已调信号中的功率。
4. 若将一个高斯噪声加到该已调信号上，使得 SNR 为 10 dB，求噪声功率。

题　解

1. $u(t)=m(t)c(t)$

$$=\begin{cases}\text{sinc}(100t)\cos(500t), & |t|\leqslant 0.1\\ 0, & \text{其余 } t\end{cases} \tag{3.2.12}$$

$$=\text{sinc}(100t)\Pi(5t)\cos(500t) \tag{3.2.13}$$

2. $m(t)$和 $u(t)$的频谱如图 3.3 所示。由图 3.3 可见，该消息信号几乎是一个带限信号，带宽为 50 Hz。

3. 已调信号的功率是消息信号功率的一半，消息信号的功率为:

$$P_m=\frac{1}{0.2}\int_{-0.1}^{0.1}\text{sinc}^2(100t)\,\mathrm{d}t$$

这个积分可用 MATLAB m 文件 quad8.m 计算，得出 $P_m=0.0495$，所以 $P_u=0.0247$。

4. 现在有

$$10\log_{10}\left(\frac{P_R}{P_n}\right)=10\Rightarrow P_n=0.1P_R=0.1P_u=0.002\,47$$

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 3.2.
% Matlab demonstration script for DSB-AM modulation. The message signal
% is m(t)=sinc(100t).
echo on
t0=.2;                              % signal duration
ts=0.001;                           % sampling interval
fc=250;                             % carrier frequency
snr=20;                             % SNR in dB (logarithmic)
fs=1/ts;                            % sampling frequency
df=0.3;                             % required freq. resolution
t=[-t0/2:ts:t0/2];                  % time vector
snr_lin=10^(snr/10);                % linear SNR
m=sinc(100*t);                      % the message signal
c=cos(2*pi*fc.*t);                  % the carrier signal
u=m.*c;                             % the DSB-AM modulated signal
[M,m,df1]=fftseq(m,ts,df);          % Fourier transform
M=M/fs;                             % scaling
```

```
[U,u,df1]=fftseq(u,ts,df);                      % Fourier transform
U=U/fs;                                         % scaling
f=[0:df1:df1*(length(m)-1)]-fs/2;               % frequency vector
signal_power=spower(u(1:length(t)));            % Compute modulated signal power.
noise_power=signal_power/snr_lin;               % Compute noise power.
noise_std=sqrt(noise_power);                    % Compute noise standard deviation.
noise=noise_std*randn(1,length(u));             % Generate noise sequence.
r=u+noise;                                      % Add noise to the modulated signal.
[R,r,df1]=fftseq(r,ts,df);                      % Fourier transform
R=R/fs;                                         % scaling
pause  % Press a key to show the modulated signal power.
signal_power
pause  %Press any key to see a plot of the message.
clf
subplot(2,2,1)
plot(t,m(1:length(t)))
xlabel('Time')
title('The message signal')
pause % Press any key to see a plot of the carrier.
subplot(2,2,2)
plot(t,c(1:length(t)))
xlabel('Time')
title('The carrier')
pause  % Press any key to see a plot of the modulated signal.
subplot(2,2,3)
plot(t,u(1:length(t)))
xlabel('Time')
title('The modulated signal')
pause   % Press any key to see a plot  of the magnitude of the message and the
        % modulated signal in the frequency domain.
subplot(2,1,1)
plot(f,abs(fftshift(M)))
xlabel('Frequency')
title('Spectrum of the message signal')
subplot(2,1,2)
plot(f,abs(fftshift(U)))
title('Spectrum of the modulated signal')
xlabel('Frequency')
pause  % Press a key to see a noise sample.
subplot(2,1,1)
plot(t,noise(1:length(t)))
title('Noise sample')
xlabel('Time')
pause  % Press a key to see the modulated signal and noise.
subplot(2,1,2)
plot(t,r(1:length(t)))
title('Signal and noise')
xlabel('Time')
pause  % Press a key to see the modulated signal and noise in freq. domain.
subplot(2,1,1)
plot(f,abs(fftshift(U)))
title('Signal spectrum')
xlabel('Frequency')
subplot(2,1,2),
plot(f,abs(fftshift(R)))
title('Signal and noise spectrum')
xlabel('Frequency')
```

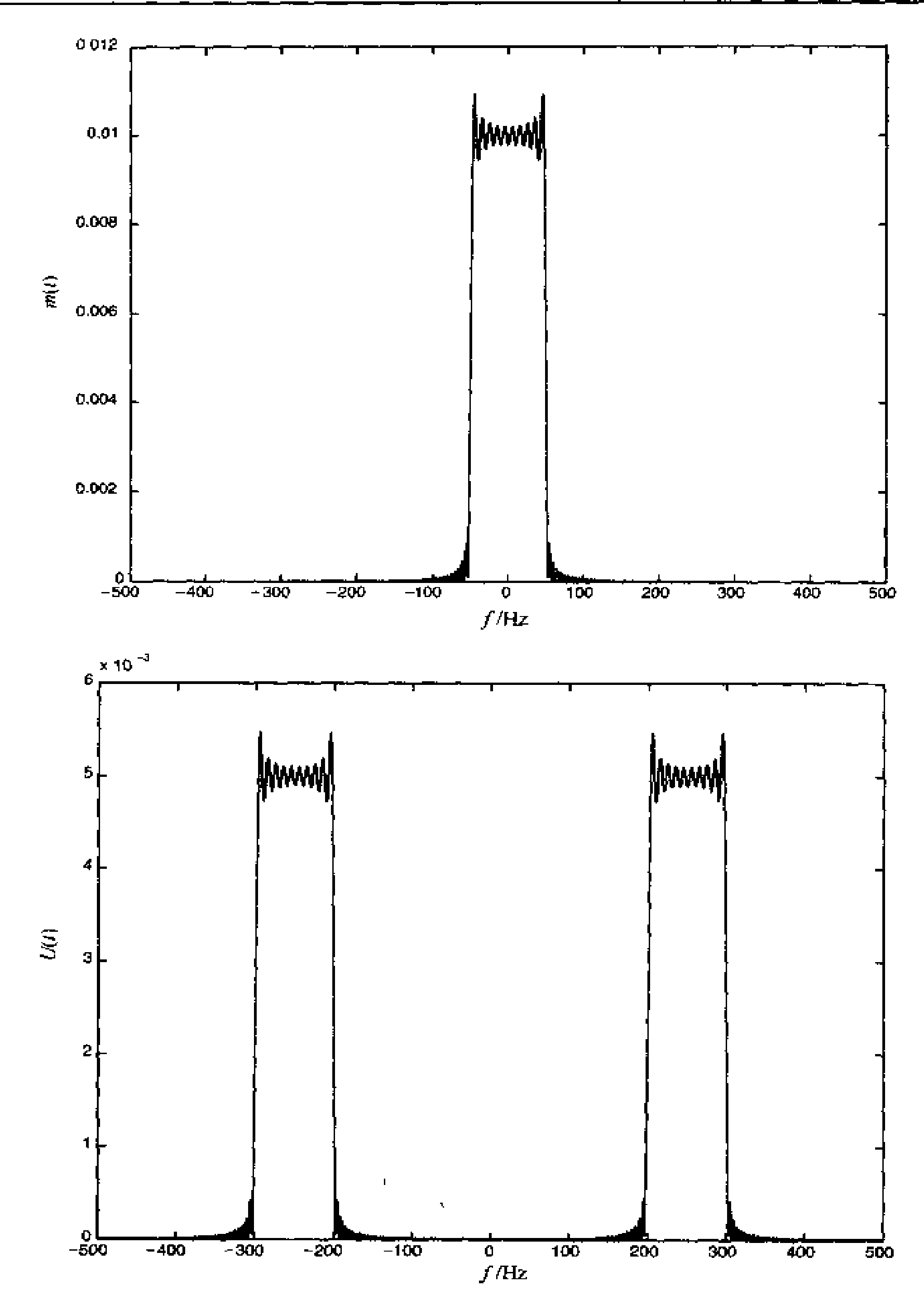


图 3.3 解说题 3.2 中的消息信号和已调信号的频谱

思考题

如果改变消息信号的持续期 t_0，会产生什么结果？特别是在大的 t_0 和小的 t_0 时会有什么结果？在带宽和信号功率上有什么效果？

下面给出的 m 文件 dsb _ mod.m 是以向量 $\boldsymbol{m}$ 给出的消息信号对频率为 f_c 的载波进行的通用 DSB 调制器。

m文件

```
function u=dsb_mod(m,ts,fc)
%           u=dsb_mod(m,ts,fc)
%DSB_MOD    takes signal m sampled at ts and carrier
%           freq. fc as input and returns the DSB modulated
%           signal. ts << 1/2fc. The modulated signal
%           is normalized to have half the message power.
%           The message signal starts at 0.

t=[0:length(m)-1]*ts;
u=m.*cos(2*pi*t);
```

3.2.2 常规 AM

在很多方面,常规 AM 和 DSB-AM 是类似的,惟一的差别是在常规 AM 中 $m(t)$ 被 $[1+am_n(t)]$ 代替,这里 $m_n(t)$ 是经归一化后的消息信号(也就是说,$|m_n(t)|\leqslant 1$)。a 是**调制指数**,而且是位于 0 和 1 之间的一个正常数。由此有:

$$u(t)=A_c[1+am_n(t)]\cos(2\pi f_c t) \tag{3.2.14}$$

和

$$U(f)=\frac{A_c}{2}[\delta(f-f_c)+aM_n(f-f_c)+\delta(f+f_c)+aM_n(f+f_c)] \tag{3.2.15}$$

将消息信号幅度加权并对其加一个常数的净效果是使 $[1+am_n(t)]$ 这一项总是正的。这就使得这些信号的解调变得十分容易,使用包络检波器就能完成。注意在 $U(f)$ 中频率为 f_c 的正弦分量的存在,这意味着发射功率中有相当一部分是在信号载波上,而信号载波其实并没有用于信息的传输。

这一点表明,与 DSB-AM 相比,常规 AM 在功率利用率上是一种不够经济的调制方法。当然,带宽等于 DSB-AM 的带宽,为

$$B_T=2W \tag{3.2.16}$$

图 3.4 显示的是消息信号和相应的常规 AM 信号的典型频域图。

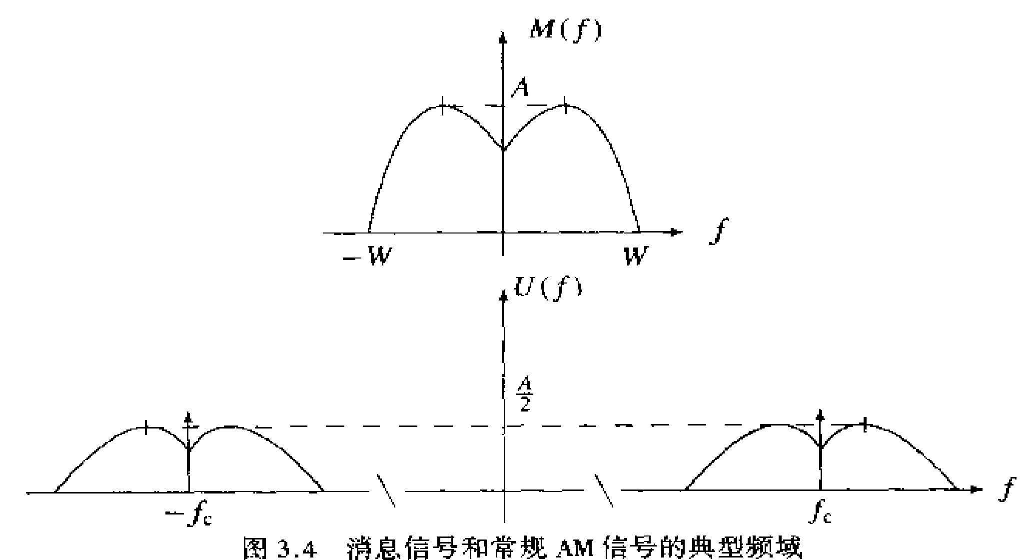


图 3.4 消息信号和常规 AM 信号的典型频域

假设消息信号是零均值信号，已调信号的功率就为：

$$P_u = \frac{A_c^2}{2}[1 + a^2 P_{m_n}] \tag{3.2.17}$$

它由两部分构成：$A_c^2/2$ 是载波的功率；$(A_c^2/2)a^2 P_{m_n}$ 是已调信号中携带消息部分的功率，而这个功率才是真正用于传送消息的功率。用于传送消息的功率与已调信号总功率的比称为**调制效率**，定义为：

$$\eta = \frac{a^2 P_{m_n}}{1 + a^2 P_{m_n}} \tag{3.2.18}$$

因为 $|m_n(t)| \leqslant 1$ 和 $a \leqslant 1$，所以总有 $\eta \leqslant 0.5$。然而，实际上 η 的值大约在 0.1 左右。信噪比为：

$$\left(\frac{S}{N}\right)_o = \eta \frac{P_R}{N_0 W} \tag{3.2.19}$$

其中 η 是调制效率。可见，与 DSB-AM 相比，SNR 减小了 η 倍。总功率中相当大的一部分是在载波上（频谱图中的冲激），而它又不携带任何信息，在接收端最终要被滤去，这就是造成性能下降的直接原因。

解说题

解说题 3.3 ［常规 AM］

消息信号 $m(t)$ 定义为：

$$m(t) = \begin{cases} 1, & 0 \leqslant t < \dfrac{t_0}{3} \\ -2, & \dfrac{t_0}{3} \leqslant t < \dfrac{2t_0}{3} \\ 0, & \text{其余 } t \end{cases}$$

用常规 AM 方法调制载波 $c(t) = \cos(2\pi f_c t)$。假设 $f_c = 250$ Hz，$t_0 = 0.15$ s，调制指数 $a = 0.85$。

1. 导出已调信号的表达式。
2. 求消息信号和已调信号的频谱。
3. 如果该消息信号是周期的，周期等于 t_0，求已调信号的功率和调制效率。
4. 如果一个噪声信号加到该消息信号上，使解调器输出端的 SNR 是 10 dB，求该噪声信号的功率。

题解

1. 首先要注意，最大 $|m(t)| = 2$，因此 $m_n(t) = m(t)/2$。由此有：

$$\begin{aligned} u(t) &= \left[1 + 0.85\frac{m(t)}{2}\right]\cos(2\pi f_c t) \\ &= \left[1 + 0.425\Pi\left(\frac{t - 0.025}{0.05}\right) - 0.85\Pi\left(\frac{t - 0.075}{0.05}\right)\right]\cos(500\pi t) \end{aligned}$$

该消息信号和已调信号如图 3.5 所示。

2. 对消息信号有

$$\mathcal{F}[m(t)] = 0.05e^{-0.05j\pi f}\text{sinc}(0.05f)(1 - 2e^{-0.1j\pi f}) \tag{3.2.20}$$

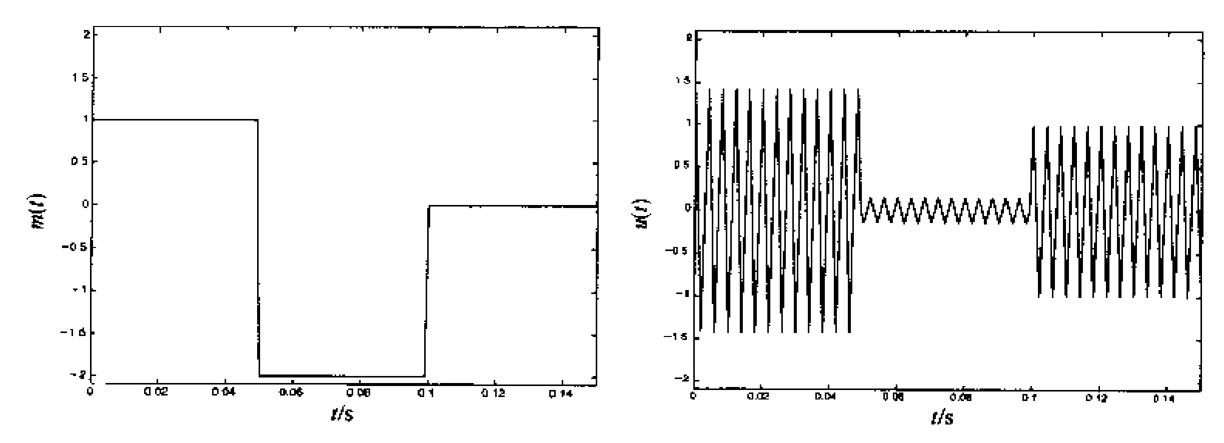


图 3.5 解说题 3.3 中的消息信号和已调信号

对已调信号有

$$U(f)=0.010\,625e^{-0.05j\pi(f-250)}\mathrm{sinc}[0.05(f-250)](1-2e^{-0.1j\pi(f-250)})$$
$$+0.010\,625e^{-0.05j\pi(f+250)}\mathrm{sinc}[0.05(f+250)](1-2e^{-0.1j\pi(f+250)})$$

图 3.6 显示的是消息信号和已调信号的频谱。

注意:这两个频谱图上的标尺不同。在已调信号的频谱上明显有两个冲激函数存在。

3. 消息信号功率可求得为:

$$P_m=\frac{1}{0.15}\left[\int_0^{0.05}\mathrm{d}t+4\int_{0.05}^{0.1}\mathrm{d}t\right]=1.667$$

归一化的消息信号功率 P_{m_n} 为:

$$P_{m_n}=\frac{1}{4}P_m=\frac{1.66}{4}=0.4167$$

调制效率是:

$$\eta=\frac{a^2P_{m_n}}{1+a^2P_{m_n}}=\frac{0.85^2\times0.4167}{1+0.85^2\times0.4167}=0.2314$$

已调信号的功率由下式给出(E 为期望值):

$$\begin{aligned}P_u&=\frac{A_c^2}{2}E[1+am_n(t)]^2\\&=\frac{1}{2}\left(1+0.3010-1.7\times\frac{0.025}{0.15}\right)\\&=0.5088\end{aligned}$$

4. 这时

$$10\log_{10}\left[\eta\left(\frac{P_R}{N_0W}\right)\right]=10$$

或者

$$\eta\left(\frac{P_R}{P_n}\right)=10$$

将 $\eta=0.2314$ 和 $P_R=P_u=0.5088$ 代入,得到:

$$P_n=\frac{\eta P_u}{10}=0.0118$$

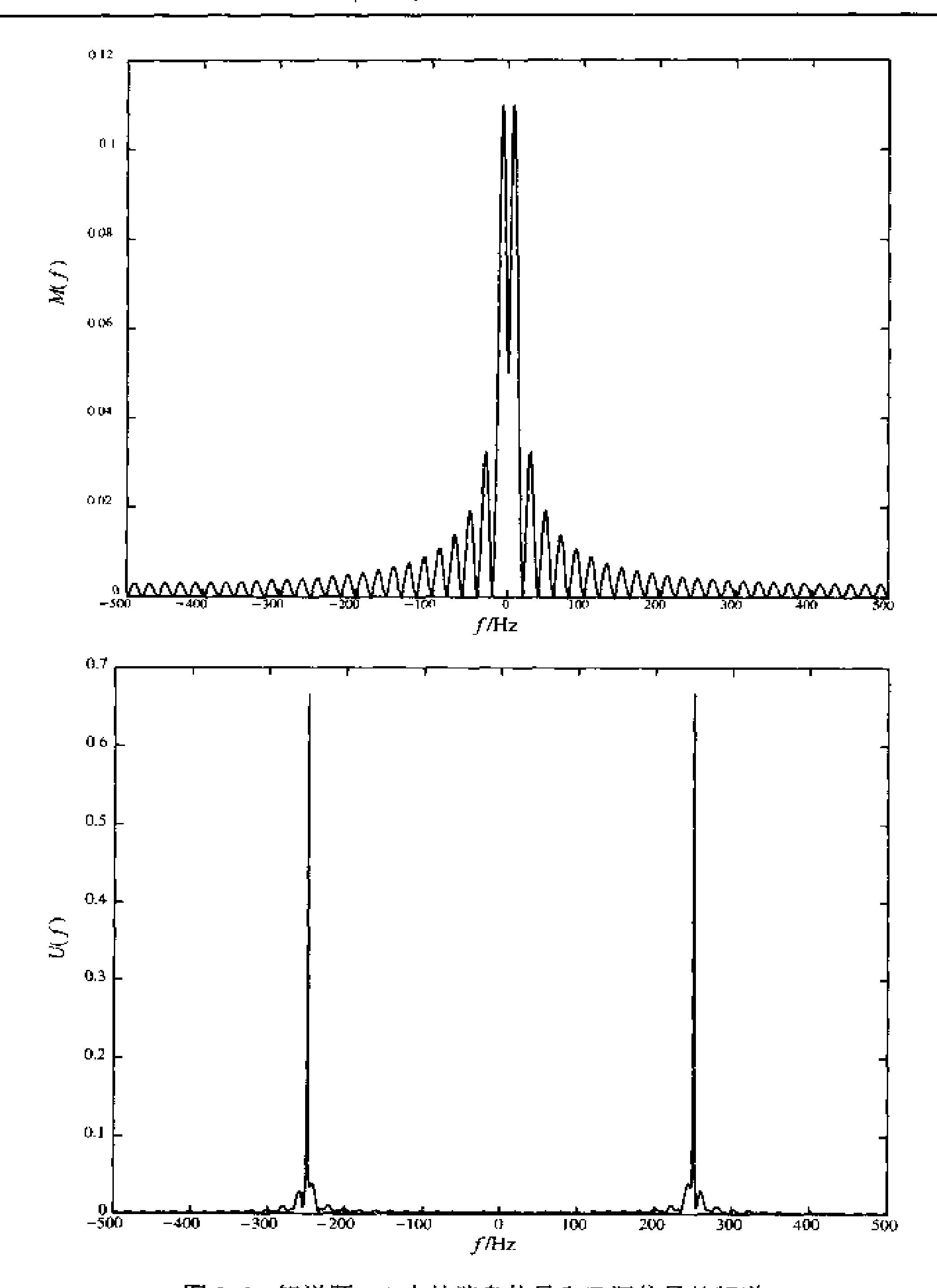


图 3.6 解说题 3.3 中的消息信号和已调信号的频谱

注 释

这个解说题求已调信号的功率时不能用下面的关系：

$$P_u = \frac{A_c^2}{2}[1 + a^2 P_{m_n}]$$

因为这里的 $m(t)$不是零均值的信号。

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 3.3.
% Demonstration script for DSB-AM modulation. The message signal
% is +1 for 0 < t < t0/3, -2 for t0/3 < t < 2t0/3, and zero otherwise.
echo on
t0=.15;                                   % signal duration
ts=0.001;                                 % sampling interval
fc=250;                                   % carrier frequency
snr=10;                                   % SNR in dB (logarithmic)
a=0.85;                                   % modulation index
fs=1/ts;                                  % sampling frequency
t=[0:ts:t0];                              % time vector
df=0.2;                                   % required frequency resolution
snr_lin=10^(snr/10);                      % SNR
% message signal
m=[ones(1,t0/(3*ts)),-2*ones(1,t0/(3*ts)),zeros(1,t0/(3*ts)+1)];
c=cos(2*pi*fc.*t);                        % carrier signal
m_n=m/max(abs(m));                        % normalized message signal
[M,m,df1]=fftseq(m,ts,df);                % Fourier transform
M=M/fs;                                   % scaling
f=[0:df1:df1*(length(m)-1)]-fs/2;         % frequency vector
u=(1+a*m_n).*c;                           % modulated signal
[U,u,df1]=fftseq(u,ts,df);                % Fourier transform
U=U/fs;                                   % scaling
signal_power=spower(u(1:length(t)));      % power in modulated signal
% power in normalized message
pmn=spower(m(1:length(t)))/(max(abs(m)))^2;
eta=(a^2*pmn)/(1+a^2*pmn);                % modulation efficiency
noise_power=eta*signal_power/snr_lin;     % noise power
noise_std=sqrt(noise_power);              % noise standard deviation
noise=noise_std*randn(1,length(u));       % Generate noise.
r=u+noise;                                % Add noise to the modulated signal.
[R,r,df1]=fftseq(r,ts,df);                % Fourier transform
R=R/fs;                                   % scaling
pause  % Press a key to show the modulated signal power.
signal_power
pause  % Press a key to show the modulation efficiency.
eta
pause  % Press any key to see a plot of the message.
subplot(2,2,1)
plot(t,m(1:length(t)))
axis([0 0.15 -2.1 2.1])
xlabel('Time')
title('The message signal')
pause
pause  % Press any key to see a plot of the carrier.
subplot(2,2,2)
plot(t,c(1:length(t)))
axis([0 0.15 -2.1 2.1])
xlabel('Time')
title('The carrier')
pause  % Press any key to see a plot of the modulated signal.
subplot(2,2,3)
plot(t,u(1:length(t)))
axis([0 0.15 -2.1 2.1])
xlabel('Time')
title('The modulated signal')
pause  % Press any key to see plots of the magnitude of the message and the
       % modulated signal in the frequency domain.
```

```
subplot(2,1,1)
plot(f,abs(fftshift(M)))
xlabel('Frequency')
title('Spectrum of the message signal')
subplot(2,1,2)
plot(f,abs(fftshift(U)))
title('Spectrum of the modulated signal')
xlabel('Frequency')
pause  % Press a key to see a noise sample.
subplot(2,1,1)
plot(t,noise(1:length(t)))
title('Noise sample')
xlabel('Time')
pause  % Press a key to see the modulated signal and noise.
subplot(2,1,2)
plot(t,r(1:length(t)))
title('Signal and noise')
xlabel('Time')
pause  % Press a key to see the modulated signal and noise in freq. domain.
subplot(2,1,1)
plot(f,abs(fftshift(U)))
title('Signal spectrum')
xlabel('Frequency')
subplot(2,1,2)
plot(f,abs(fftshift(R)))
title('Signal and noise spectrum')
xlabel('Frequency')
```

下面给出的 MATLAB m 文件 am_mode.m 是一种通用的常规 AM 调制器。

m 文件

```
function u=am_mod(a,m,ts,fc)
%          u=am_mod(a,m,ts,fc)
%AM_MOD    takes signal m sampled at ts and carrier
%          freq. fc as input and returns the AM modulated
%          signal. "a" is the modulation index
%          and ts << 1/2fc.

t=[0:length(m)-1]*ts;
c=cos(2*pi*fc.*t);
m_n=m/max(abs(m));
u=(1+a*m_n).*c;
```

3.2.3 SSB-AM

除掉 DSB-AM 的一个边带就可以得到 SSB-AM，因此 SSB-AM 占有 DSB-AM 一半的带宽。依据所保留的边带是上边带还是下边带，就有两种类型的 SSB-AM：USSB-AM 和 LSSB-AM。这些信号的时域表示为：

$$u(t)=\frac{A_c}{2}m(t)\cos(2\pi f_c t)\mp\frac{A_c}{2}\hat{m}(t)\sin(2\pi f_c t) \tag{3.2.21}$$

其中减号对应于 USSB-AM，加号对应于 LSSB-AM，$\hat{m}(t)$是 $m(t)$的希尔伯特变换，定义为 $\hat{m}(t)$

$= m(t) * (1/(\pi t))$,在频域中则是 $\hat{M}(f) = -\mathrm{j}\,\mathrm{sgn}(f) M(f)$。换句话说,一个信号的希尔伯特变换代表在全部信号分量上进行 $\pi/2$ 的相移。这样,在频域中有

$$U_{\mathrm{USSB}}(f) = \begin{cases} [M(f-f_c) + M(f+f_c)], & f_c \leqslant |f| \\ 0, & 其余\ f \end{cases} \tag{3.2.22}$$

和

$$U_{\mathrm{LSSB}}(f) = \begin{cases} [M(f-f_c) + M(f+f_c)], & |f| \leqslant f_c \\ 0, & 其余\ f \end{cases} \tag{3.2.23}$$

图 3.7 显示的是某个消息信号及其对应的 USSB-AM 已调信号的典型频谱图。

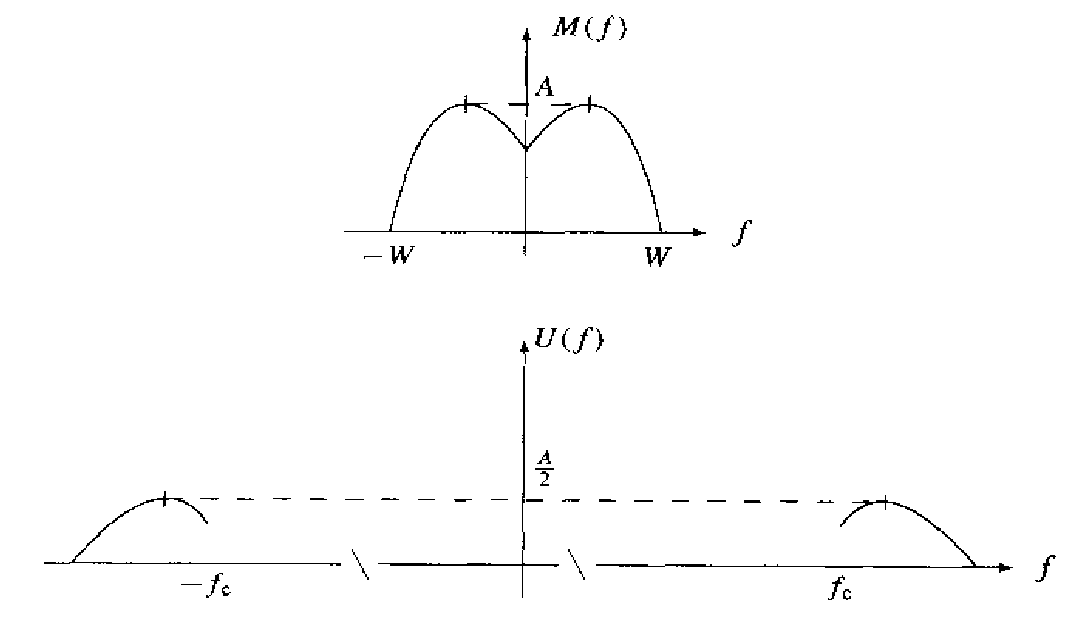


图 3.7 消息信号及其对应的 USSB-AM 已调信号的典型频谱

SSB 信号的带宽是 DSB 和常规 AM 带宽的一半,所以等于消息信号的带宽,即

$$B_{\mathrm{T}} = W \tag{3.2.24}$$

SSB 信号的功率为

$$P_u = \frac{A_c^2}{4} P_m \tag{3.2.25}$$

注意,由于已经除去了一个边带,所以 SSB 的功率是对应的 DSB-AM 的一半。另一方面,因为这个已调信号只有对应的 DSB-AM 信号带宽的一半,所以在接收机前端点上的噪声功率也是其相应的 DSB-AM 情况下的一半,因此在两种系统中 SNR 是相同的,即

$$\left(\frac{S}{N}\right)_o = \frac{P_R}{N_0 W} \tag{3.2.26}$$

解说题

解说题 3.4 [单边带举例]

有消息信号 $m(t)$ 为:

$$m(t)=\begin{cases}1, & 0\leqslant t<\frac{t_0}{3}\\ -2, & \frac{t_0}{3}\leqslant t<\frac{2t_0}{3}\\ 0, & \text{其余 } t\end{cases}$$

用 LSSB-AM 方法调制载波 $c(t)=\cos(2\pi f_c t)$。假定 $t_0=0.15\ \mathrm{s}, f_c=250\ \mathrm{Hz}$。

1. 画出消息信号的希尔伯特变换和已调信号 $u(t)$,同时也画出已调信号的频谱。

2. 假设消息信号是周期的,周期为 t_0,求已调信号的功率。

3. 若一个噪声加在已调信号上,使解调后的 SNR 是 10 dB,求噪声功率。

题 解

1. 消息信号的希尔伯特变换可用 MATLAB 的希尔伯特变换 m 文件 hilbert.m 计算得出。然而,值得注意的是,这个函数所得到的是一个复数序列,其实部是原序列,而虚部才是要求的希尔伯特变换。因此,序列 m 的希尔伯特变换用指令 image(hilbert(m))得到。现在,利用下面的关系:

$$u(t)=m(t)\cos(2\pi f_c t)+\hat{m}(t)\sin(2\pi f_c t) \tag{3.2.27}$$

就能求得已调信号。图 3.8 显示的是 $\hat{m}(t)$的图和 LSSB-AM 已调信号 $u(t)$的频谱图。

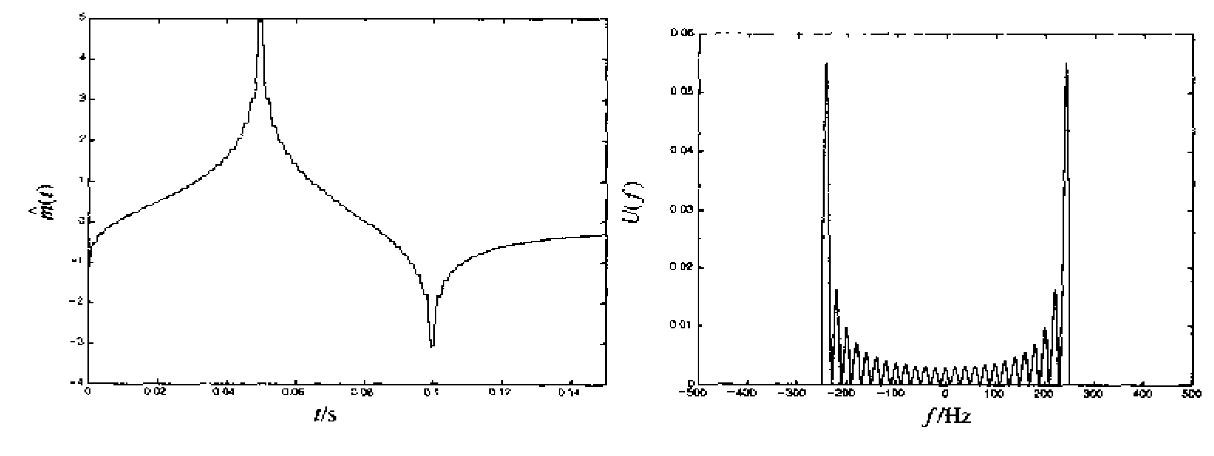


图 3.8 $m(t)$的希尔伯特变换和 LSSB-AM 已调信号的频谱

2. 消息信号的功率是:

$$P_m=\frac{1}{0.15}\int_0^{0.15}m^2(t)\mathrm{d}t=1.667$$

因此有:

$$P_u=\frac{A_c^2}{4}P_m=0.416$$

3. 解调后的 SNR 由下式给出:

$$10\log_{10}\left(\frac{P_R}{P_n}\right)_o=10$$

所以

$$P_n=0.1P_R=0.1P_u=0.0416$$

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 3.4.
% Demonstration script for LSSB-AM modulation. The message signal
% is +1 for 0 < t < t0/3, -2 for t0/3 < t < 2t0/3, and zero otherwise.
echo on
t0=.15;                              % signal duration
ts=0.001;                            % sampling interval
fc=250;                              % carrier frequency
snr=10;                              % SNR in dB (logarithmic)
fs=1/ts;                             % sampling frequency
df=0.25;                             % desired freq. resolution
t=[0:ts:t0];                         % time vector
snr_lin=10^(snr/10);                 % SNR
% the message vector
m=[ones(1,t0/(3*ts)),-2*ones(1,t0/(3*ts)),zeros(1,t0/(3*ts)+1)];
c=cos(2*pi*fc.*t);                   % carrier vector
udsb=m.*c;                           % DSB modulated signal
[UDSB,udssb,df1]=fftseq(udsb,ts,df); % Fourier transform
UDSB=UDSB/fs;                        % scaling
f=[0:df1:df1*(length(udssb)-1)]-fs/2; % frequency vector
n2=ceil(fc/df1);                     % location of carrier in freq. vector
% Remove the upper sideband from DSB.
UDSB(n2:length(UDSB)-n2)=zeros(size(UDSB(n2:length(UDSB)-n2)));
ULSSB=UDSB;                          % Generate LSSB-AM spectrum.
[M,m,df1]=fftseq(m,ts,df);           % Fourier transform
M=M/fs;                              % scaling
u=real(ifft(ULSSB))*fs;              % Generate LSSB signal from spectrum.
signal_power=spower(udsb(1:length(t)))/2;
%                                    % Compute signal power.
noise_power=signal_power/snr_lin;    % Compute noise power.
noise_std=sqrt(noise_power);         % Compute noise standard deviation.
noise=noise_std*randn(1,length(u));  % Generate noise vector.
r=u+noise;                           % Add the signal to noise.
[R,r,df1]=fftseq(r,ts,df);           % Fourier transform
R=R/fs;                              % scaling
pause  % Press a key to show the modulated signal power.
signal_power
pause  % Press any key to see a plot of the message signal.
clf
subplot(2,1,1)
plot(t,m(1:length(t)))
axis([0, 0.15, -2.1, 2.1])
xlabel('Time')
title('The message signal')
pause  % Press any key to see a plot of the carrier.
subplot(2,1,2)
plot(t,c(1:length(t)))
xlabel('Time')
title('The carrier')
pause  % Press any key to see a plot of the modulated signal and its spectrum.
clf
subplot(2,1,1)
plot([0:ts:ts*(length(u)-1)/8],u(1:length(u)/8))
xlabel('Time')
```

```
title('The LSSB-AM modulated signal')
subplot(2,1,2)
plot(f,abs(fftshift(ULSSB)))
xlabel('Frequency')
title('Spectrum of the LSSB-AM modulated signal')
pause   % Press any key to see the spectra of the message and the modulated signals.
clf
subplot(2,1,1)
plot(f,abs(fftshift(M)))
xlabel('Frequency')
title('Spectrum of the message signal')
subplot(2,1,2)
plot(f,abs(fftshift(ULSSB)))
xlabel('Frequency')
title('Spectrum of the LSSB-AM modulated signal')

pause   % Press any key to see a noise sample.
subplot(2,1,1)
plot(t,noise(1:length(t)))
title('Noise sample')
xlabel('Time')
pause   % Press a key to see the modulated signal and noise.
subplot(2,1,2)
plot(t,r(1:length(t)))
title('Modulated signal and noise')
xlabel('Time')
subplot(2,1,1)
pause % Press any key to see the spectrum of the modulated signal.
plot(f,abs(fftshift(ULSSB)))
title('Modulated signal spectrum')
xlabel('Frequency')
subplot(2,1,2)


pause   % Press a key to see the modulated signal noise in freq. domain.
plot(f,abs(fftshift(R)))
title('Modulated signal noise spectrum')
xlabel('Frequency')
```

下面给出的 m 文件 ussb _ mod.m 和 lssb _ mod.m 是以向量 ***m*** 给出的消息信号使用 USSB 和 LSSB 调制方法进行的调制。

m 文件

```
function u=ussb_mod(m,ts,fc)
%              u=ussb_mod(m,ts,fc)
%USSB_MOD      takes signal m sampled at ts and carrier
%              freq fc as input and returns the USSB modulated
%              signal. ts << 1/2fc.
t=[0:length(m)-1]*ts;
u=m.*cos(2*pi*t)-imag(hilbert(m)).*sin(2*pi*t);
```

m文件

```
function u=lssb_mod(m,ts,fc)
%               u=lssb_mod(m,ts,fc)
%LSSB_MOD       takes signal m sampled at ts and carrier
%               freq. fc as input and returns the LSSB modulated
%               signal. ts << 1/2fc.
t=[0:length(m)-1]*ts;
u=m.*cos(2*pi*t)+imag(hilbert(m)).*sin(2*pi*t);
```

3.3 AM信号的解调

解调是从已调信号中提取消息信号的过程。解调过程取决于所使用的调制方式。对于DSB-AN来说,解调的方法是相干解调,它要求在接收端有一个与载波同频同相位的信号存在,对于常规AM,用包络检波器解调,这时关于载波频率和相位的精确了解在接收端已不太重要了,所以解调十分容易。对于DSB-AM和SSB-AM的相干解调,由如下部分组成:用一个与载波同频和同相位的正弦信号乘以已调信号(即混频),然后将乘积通过一个低通滤波器来完成。在接收端产生的所要求的正弦振荡的振荡器称为**本地振荡器**。

3.3.1 DSB-AM解调

在DSB中已调信号由 $A_c m(t)\cos(2\pi f_c t)$ 给出,当将它乘以 $\cos(2\pi f_c t)$ 或者说与 $\cos(2\pi f_c t)$ 混频以后,就得到了:

$$y(t)=A_c m(t)\cos(2\pi f_c t)\cos(2\pi f_c t)=\frac{A_c}{2}m(t)+\frac{A_c}{2}m(t)\cos(4\pi f_c t) \qquad (3.3.1)$$

其中 $y(t)$ 为混频器输出,而它的傅里叶变换由下式给出:

$$Y(f)=\frac{A_c}{2}M(f)+\frac{A_c}{4}M(f-2f_c)+\frac{A_c}{4}M(f+2f_c) \qquad (3.3.2)$$

由上可知,混频器输出中有一个低频分量 $(A_c/2)M(f)$ 和 $\pm 2f_c$ 附近的高频分量。当 $y(t)$ 通过带宽为 W 的低通滤波器时,高频分量被滤去了,而正比于消息信号的低频分量 $\left(\left(\frac{A_c}{2}\right)m(t)\right)$ 被调制出。这个过程如图3.9所示。

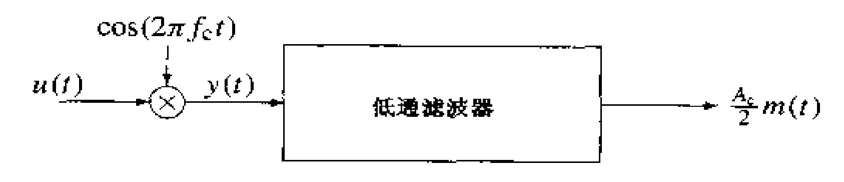


图3.9 DSB-AM信号的解调

解说题

解说题3.5 [DSB-AM解调]

消息信号 $m(t)$ 定义为:

$$m(t)=\begin{cases}1, & 0\leqslant t<\dfrac{t_0}{3}\\ -2, & \dfrac{t_0}{3}\leqslant t<\dfrac{2t_0}{3}\\ 0, & \text{其余 } t\end{cases}$$

用该消息信号以 DSB-AM 方式调制载波 $c(t)=\cos(2\pi f_c t)$，所得已调信号为 $u(t)$。假设 $t_0=0.15$ s，$f_c=250$ Hz。

1. 求 $u(t)$ 的表达式。
2. 求 $m(t)$ 和 $u(t)$ 的频谱。
3. 将这个已调信号解调，并恢复 $m(t)$，画出在时域和频域内的结果。

题 解

1. 本题的前两部分与解说题 3.1 的前两部分是相同的，这里只需要重复这些结果：

$$u(t)=\left[\Pi\left(\frac{t-0.025}{0.05}\right)-2\Pi\left(\frac{t-0.075}{0.05}\right)\right]\cos(500\pi t)$$

$$\begin{aligned}\mathcal{F}[m(t)]&=\frac{t_0}{3}e^{-j\pi f t_0/3}\operatorname{sinc}\left(\frac{t_0 f}{3}\right)-\frac{2t_0}{3}e^{-j\pi f t_0}\operatorname{sinc}\left(\frac{t_0 f}{3}\right)\\&=\frac{t_0}{3}e^{-j\pi f t_0/3}\operatorname{sinc}\left(\frac{t_0 f}{3}\right)(1-2e^{-j2\pi t_0 f/3})\\&=0.05e^{-0.05j\pi f}\operatorname{sinc}(0.05f)(1-2e^{-0.01j\pi f})\end{aligned}$$

因此

$$\begin{aligned}U(f)=&\,0.025e^{-0.05j\pi(f-250)}\operatorname{sinc}[0.05(f-250)](1-2e^{-0.1j\pi(f-250)})\\&+0.025e^{-0.05j\pi(f+250)}\operatorname{sinc}[0.05(f+250)](1-2e^{-0.1j\pi(f+250)})\end{aligned}$$

2. 为了解调，要将 $u(t)$ 乘以 $\cos(2\pi f_c t)$，得到混频器输出 $y(t)$：

$$\begin{aligned}y(t)&=u(t)\cos(2\pi f_c t)\\&=\left[\Pi\left(\frac{t-0.025}{0.05}\right)-2\Pi\left(\frac{t-0.075}{0.05}\right)\right]\cos^2(500\pi t)\\&=\frac{1}{2}\left[\Pi\left(\frac{t-0.025}{0.05}\right)-2\Pi\left(\frac{t-0.075}{0.05}\right)\right]\\&\quad+\frac{1}{2}\left[\Pi\left(\frac{t-0.025}{0.05}\right)-2\Pi\left(\frac{t-0.075}{0.05}\right)\right]\cos(1000\pi t)\end{aligned}$$

它的傅里叶变换是：

$$\begin{aligned}Y(f)=&\,0.025e^{-0.05j\pi f}\operatorname{sinc}(0.05f)(1-2e^{-0.01j\pi f})\\&+0.0125e^{-0.05j\pi(f-500)}\operatorname{sinc}[0.05(f-500)](1-2e^{-0.1j\pi(f-500)})\\&+0.0125e^{-0.05j\pi(f+500)}\operatorname{sinc}[0.05(f+500)](1-2e^{-0.1j\pi(f+500)})\end{aligned}$$

其中的第一项对应于消息信号，后两项对应于两倍载波频率的高频分量。可见将第一项滤出就产生了原消息信号（有一个比例常数）。$U(f)$ 和 $Y(f)$ 的幅度如图 3.10 所示。

如上所示，混频器输出的频谱中有一个低频分量，它是非常类似于消息信号的谱（除了因子 $\frac{1}{2}$ 外），和一个位于 $\pm 2f_c$（这时是 500 Hz）的带通分量。利用一个低通滤波器，就能将低频分量与带通分量分隔开。为了恢复消息信号 $m(t)$，现将 $y(t)$ 通过带宽为 150 Hz 的低通滤波器。

由于现在的消息信号不是严格带限的,所以这里滤波器带宽的选取多少有些任意性。对严格带限的消息信号来说,低通滤波器带宽的合适选择应是 W,即消息信号的带宽。因此,此处所用的理想低通滤波器的特性为:

$$H(f)=\begin{cases}1, & |f|\leqslant 150\\ 0, & \text{其余 } f\end{cases}$$

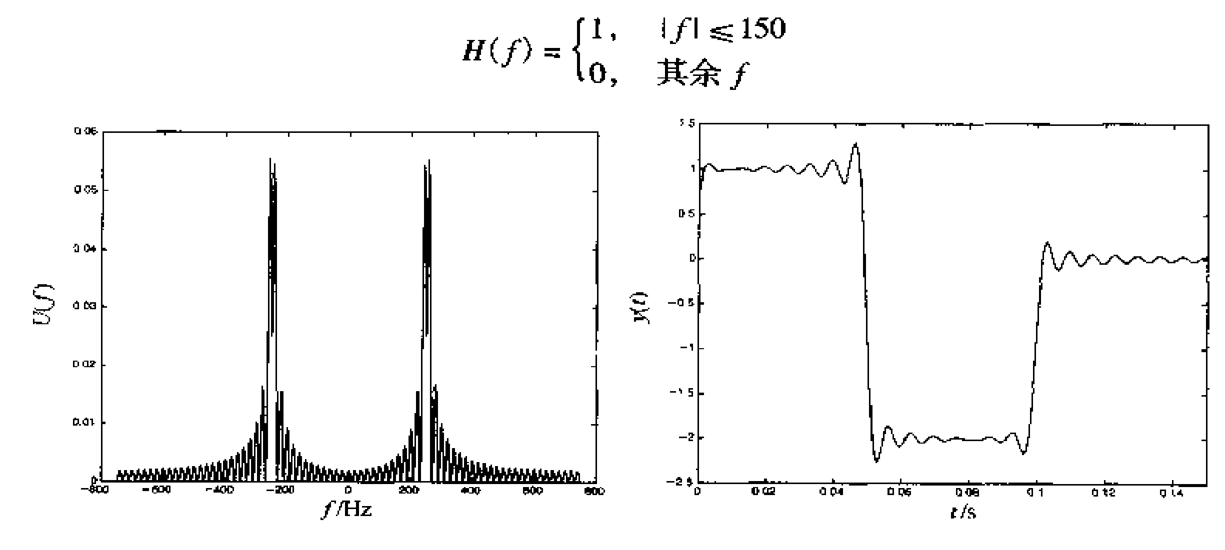


图 3.10 解说题 3.5 中的已调信号和混频器输出的谱

$m(t)$和解调器输出在频谱上的比较如图 3.11 所示,两者在时域的比较如图 3.12 所示。

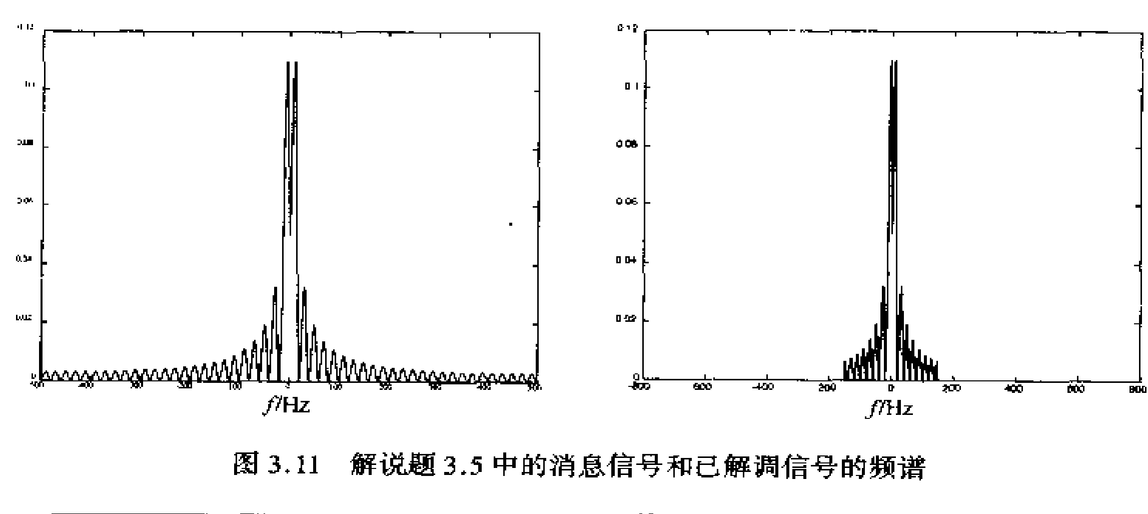


图 3.11 解说题 3.5 中的消息信号和已解调信号的频谱

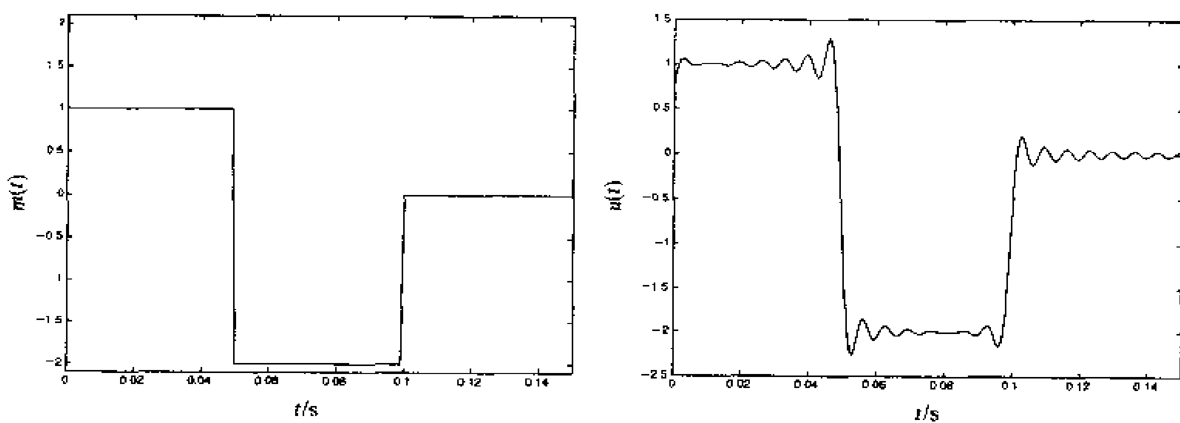


图 3.12 解说题 3.5 中的消息信号和解调器输出

本题的 MATLAB 脚本如下所示。

m文件

```
% MATLAB script for Illustrative Problem 3.5.
% Demonstration script for DSB-AM demodulation. The message signal
% is +1 for 0 < t < t0/3, -2 for t0/3 < t < 2t0/3, and zero otherwise.
echo on
t0=.15;                              % signal duration
ts=1/1500;                           % sampling interval
fc=250;                              % carrier frequency
fs=1/ts;                             % sampling frequency
t=[0:ts:t0];                         % time vector
df=0.3;                              % desired frequency resolution
% message signal
m=[ones(1,t0/(3*ts)),-2*ones(1,t0/(3*ts)),zeros(1,t0/(3*ts)+1)];
c=cos(2*pi*fc.*t);                   % carrier signal
u=m.*c;                              % modulated signal
y=u.*c;                              % mixing
[M,m,df1]=fftseq(m,ts,df);           % Fourier transform
M=M/fs;                              % scaling
[U,u,df1]=fftseq(u,ts,df);           % Fourier transform
U=U/fs;                              % scaling
[Y,y,df1]=fftseq(y,ts,df);           % Fourier transform
Y=Y/fs;                              % scaling
f_cutoff=150;                        % cutoff freq. of the filter
n_cutoff=floor(150/df1);             % Design the filter.
f=[0:df1:df1*(length(y)-1)]-fs/2;
H=zeros(size(f));
H(1:n_cutoff)=2*ones(1,n_cutoff);
H(length(f)-n_cutoff+1:length(f))=2*ones(1,n_cutoff);
DEM=H.*Y;                            % spectrum of the filter output
dem=real(ifft(DEM))*fs;              % filter output
pause % Press a key to see the effect of mixing.
clf
subplot(3,1,1)
plot(f,fftshift(abs(M)))
title('Spectrum of the Message Signal')
xlabel('Frequency')
subplot(3,1,2)
plot(f,fftshift(abs(U)))
title('Spectrum of the Modulated Signal')
xlabel('Frequency')
subplot(3,1,3)
plot(f,fftshift(abs(Y)))
title('Spectrum of the Mixer Output')
xlabel('Frequency')
pause % Press a key to see the effect of filtering on the mixer output.
clf
subplot(3,1,1)
plot(f,fftshift(abs(Y)))
title('Spectrum of the Mixer Output')
xlabel('Frequency')
subplot(3,1,2)
plot(f,fftshift(abs(H)))
title('Lowpass Filter Characteristics')
xlabel('Frequency')
subplot(3,1,3)
plot(f,fftshift(abs(DEM)))
title('Spectrum of the Demodulator output')
xlabel('Frequency')
```

```
pause % Press a key to compare the spectra of the message and the received signal.
clf
subplot(2,1,1)
plot(f,fftshift(abs(M)))
title('Spectrum of the Message Signal')
xlabel('Frequency')
subplot(2,1,2)
plot(f,fftshift(abs(DEM)))
title('Spectrum of the Demodulator Output')
xlabel('Frequency')
pause % Press a key to see the message and the demodulator output signals.
subplot(2,1,1)
plot(t,m(1:length(t)))
title('The Message Signal')
xlabel('Time')
subplot(2,1,2)
plot(t,dem(1:length(t)))
title('The Demodulator Output')
xlabel('Time')
```

解说题

解说题 3.6 [DSB-AM 解调中相位误差的后果]

在 DSB-AM 信号解调中,已经假设本地振荡器的相位等于载波的相位。如果情况不是这样,而是本地振荡器和载波之间存在某一个相移 ϕ,解调过程会如何改变?

题 解

这时有 $u(t)=A_c m(t)\cos(2\pi f_c t)$,本地振荡器产生的正弦信号为 $\cos(2\pi f_c t+\phi)$,这两个信号混频以后为:

$$y(t)=A_c m(t)\cos(2\pi f_c t)\cos(2\pi f_c t+\phi) \tag{3.3.3}$$

$$=\frac{A_c}{2}m(t)\cos\phi+\frac{A_c}{2}m(t)\cos(4\pi f_c t+\phi) \tag{3.3.4}$$

和前面相同,在混频器输出中有两项,带通项可以用低通滤波器滤掉,然而低频项 $(A_c/2)m(t)\cos\phi$ 与 ϕ 有关。低频项的功率为:

$$P_{dem}=\frac{A_c^2}{4}P_m\cos^2\phi \tag{3.3.5}$$

其中 P_m 是消息信号功率。因此可见,在这种情况下恢复出的消息信号基本上没有失真,但是在功率上由于 $\cos^2\phi$ 而有损失。若 $\phi=\pi/4$,功率损失是 3 dB;若 $\phi=\pi/2$,则在解调过程中什么也恢复不出来。

3.3.2 SSB-AM 解调

SSB-AM 信号的解调过程基本上与 DSB-AM 信号的解调过程是相同的,也就是说混频之后紧接着低通滤波,这时

$$u(t)=\frac{A_c}{2}m(t)\cos(2\pi f_c t)\mp\frac{A_c}{2}\hat{m}(t)\sin(2\pi f_c t) \tag{3.3.6}$$

其中减号对应于 USSB,加号对应于 LSSB; $u(t)$ 与本地振荡器混频之后得到输出:

$$y(t)=\frac{A_c}{2}m(t)\cos^2(2\pi f_c t)\mp\frac{A_c}{2}\hat{m}(t)\sin(2\pi f_c t)\cos(2\pi f_c t)$$

$$=\frac{A_c}{4}m(t)+\frac{A_c}{4}m(t)\cos(4\pi f_c t)\mp\frac{A_c}{4}\hat{m}(t)\sin(4\pi f_c t) \quad (3.3.7)$$

其中含有在 $\pm 2f_c$ 的带通分量和正比于消息信号的低频分量。低频分量用低通滤波器滤出，以恢复消息信号。图 3.13 显示的是对于 USSB-AM 的解调过程。

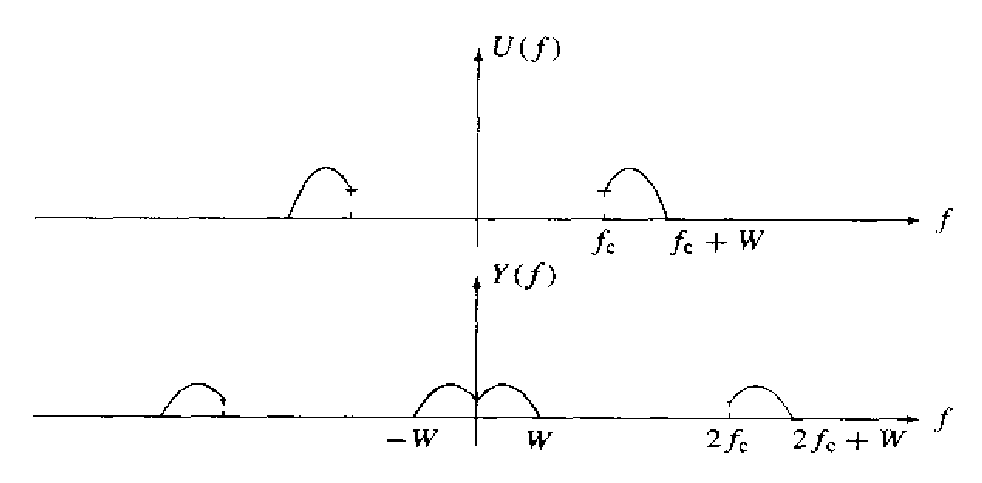


图 3.13　USSB-AM 的解调

解说题

解说题 3.7　[LSSB-AM 的例子]

在 LSSB-AM 解调系统中，若消息信号为：

$$m(t)=\begin{cases}1, & 0\leqslant t<\dfrac{t_0}{3}\\ -2, & \dfrac{t_0}{3}\leqslant t<\dfrac{2t_0}{3}\\ 0, & \text{其余 } t\end{cases}$$

其中 $t_0=0.15$ s，载波频率 $f_c=250$ Hz，求 $U(f)$ 和 $Y(f)$，并将解调信号与消息信号进行比较。

题　解

已调信号和它的频谱在解说题 3.4 中已给出，$U(f)$ 的表达式是：

$$U(f)=\begin{cases}0.025e^{-0.05j\pi(f-250)}\text{sinc}(0.05(f-250))(1-2e^{-0.1j\pi(f-250)}) & \\ +0.025e^{-0.05j\pi(f+250)}\text{sinc}(0.05(f+250))(1-2e^{-0.1j\pi(f+250)}), & |f|\leqslant f_c\\ 0, & \text{其余 } f\end{cases}$$

$$Y(f)=\frac{1}{2}U(f-f_c)+\frac{1}{2}U(f+f_c)\approx$$

$$\begin{cases}0.0125e^{-0.05j\pi f}\text{sinc}(0.05f)(1-2e^{-0.01j\pi f}), & |f|\leqslant f_c\\ 0.0125e^{-0.05j\pi(f-500)}\text{sinc}(0.05(f-500))(1-2e^{-0.01j\pi(f-500)}), & f_c\leqslant f\leqslant 2f_c\\ 0.0125e^{-0.05j\pi(f+500)}\text{sinc}(0.05(f+500))(1-2e^{-0.01j\pi(f+500)}), & -2f_c\leqslant f\leqslant -f_c\\ 0, & \text{其余 } f\end{cases}$$

图3.14显示的是$U(f)$的图。信号$y(t)$用截止频率为150 Hz的低通滤波器滤波，图3.15是输出的频谱。图3.16将原消息信号与已解调信号进行了比较。

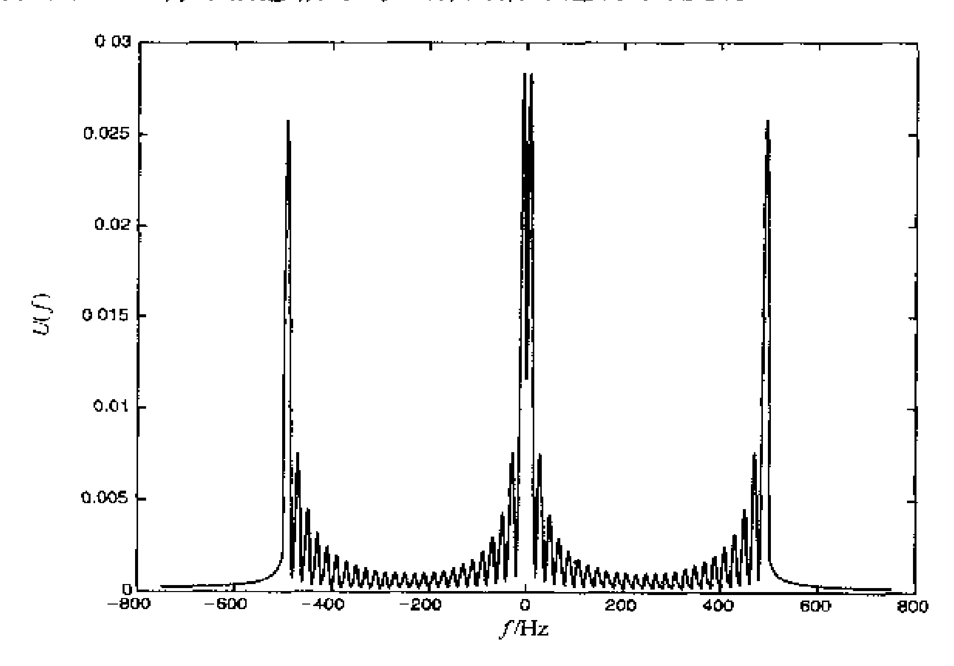


图3.14 解说题3.7中的混频器输出的幅度谱

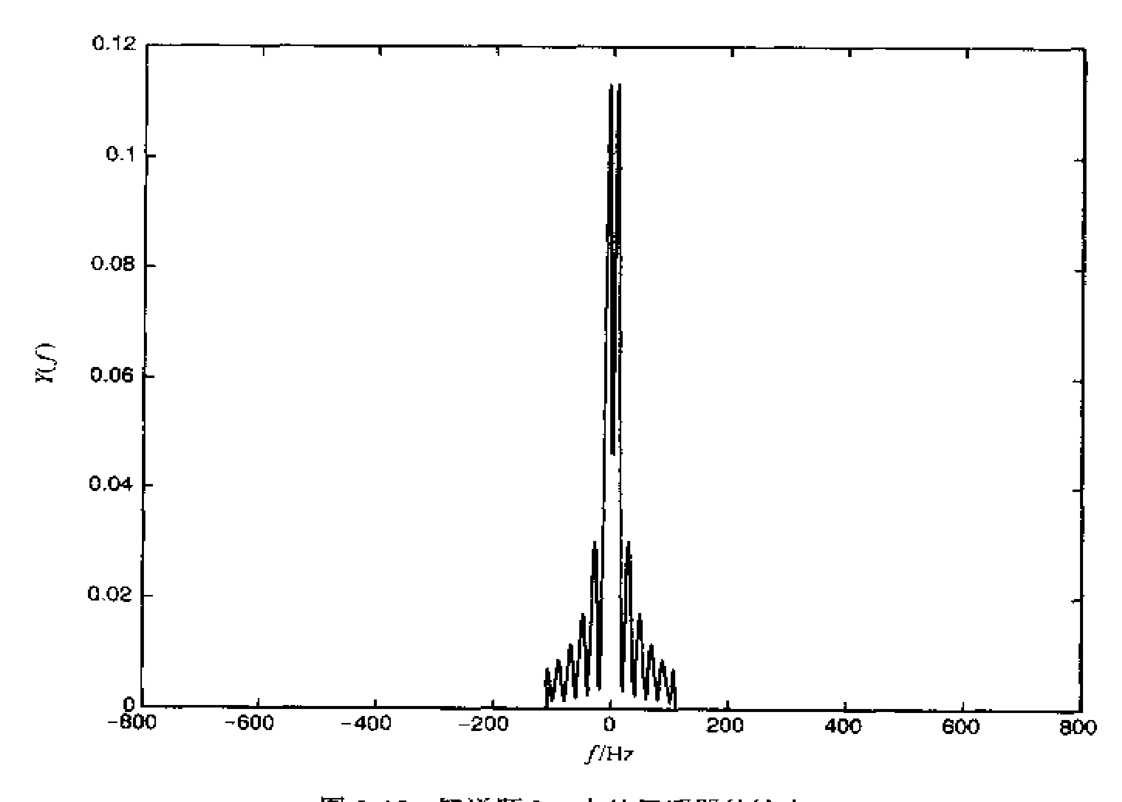


图3.15 解说题3.7中的解调器的输出

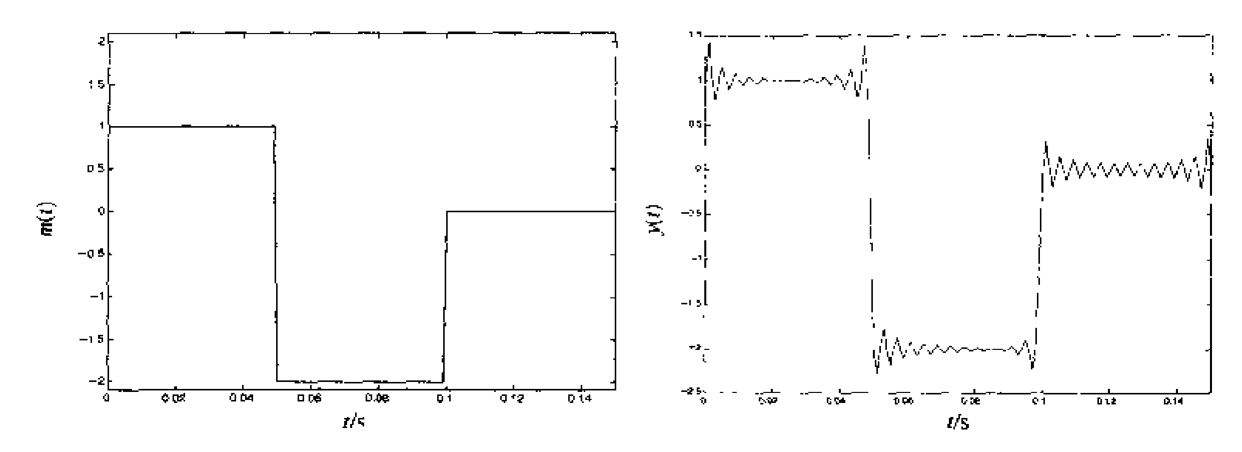


图 3.16　解说题 3.7 中的消息信号与解调器输出信号

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 3.7.
% Demonstration script for LSSB-AM demodulation. The message signal
% is +1 for 0 < t < t0/3, -2 for t0/3 < t < 2t0/3, and zero otherwise.
echo on
t0=.15;                                   % signal duration
ts=1/1500;                                % sampling interval
fc=250;                                   % carrier frequency
fs=1/ts;                                  % sampling frequency
df=0.25;                                  % desired freq.resolution
t=[0:ts:t0];                              % time vector
% the message vector
m=[ones(1,t0/(3*ts)),-2*ones(1,t0/(3*ts)),zeros(1,t0/(3*ts)+1)];
c=cos(2*pi*fc.*t);                        % carrier vector
udsb=m.*c;                                % DSB modulated signal
[UDSB,udsb,df1]=fftseq(udsb,ts,df);       % Fourier transform
UDSB=UDSB/fs;                             % scaling
n2=ceil(fc/df1);                          % location of carrier in freq. vector
% Remove the upper sideband from DSB.
UDSB(n2:length(UDSB)-n2)=zeros(size(UDSB(n2:length(UDSB)-n2)));
ULSSB=UDSB;                                % Generate LSSB-AM spectrum.
[M,m,df1]=fftseq(m,ts,df);              % spectrum of the message signal
M=M/fs;                                   % scaling
f=[0:df1:df1*(length(M)-1)]-fs/2;         % frequency vector
u=real(ifft(ULSSB))*fs;                   % Generate LSSB signal from spectrum.
% mixing
y=u.*cos(2*pi*fc*[0:ts:ts*(length(u)-1)]);
[Y,y,df1]=fftseq(y,ts,df);                % spectrum of the output of the mixer
Y=Y/fs;                                   % scaling
f_cutoff=150;                             % Choose the cutoff freq. of the filter.
n_cutoff=floor(150/df);                   % Design the filter.
H=zeros(size(f));
H(1:n_cutoff)=4*ones(1,n_cutoff);
% spectrum of the filter output
H(length(f)-n_cutoff+1:length(f))=4*ones(1,n_cutoff);
DEM=H.*Y;                                 % spectrum of the filter output
dem=real(ifft(DEM))*fs;                   % filter output
pause % Press a key to see the effect of mixing.
```

```
clf
subplot(3,1,1)
plot(f,fftshift(abs(M)))
title('Spectrum of the Message Signal')
xlabel('Frequency')
subplot(3,1,2)
plot(f,fftshift(abs(ULSSB)))
title('Spectrum of the Modulated Signal')
xlabel('Frequency')
subplot(3,1,3)
plot(f,fftshift(abs(Y)))
title('Spectrum of the Mixer Output')
xlabel('Frequency')
pause % Press a key to see the effect of filtering on the mixer output.
clf
subplot(3,1,1)
plot(f,fftshift(abs(Y)))
title('Spectrum of the Mixer Output')
xlabel('Frequency')
subplot(3,1,2)
plot(f,fftshift(abs(H)))
title('Lowpass Filter Characteristics')
xlabel('Frequency')
subplot(3,1,3)
plot(f,fftshift(abs(DEM)))
title('Spectrum of the Demodulator output')
xlabel('Frequency')
pause % Press a key to see the message and the demodulator output signals.
subplot(2,1,1)
plot(t,m(1:length(t)))
title('The Message Signal')
xlabel('Time')
subplot(2,1,2)
plot(t,dem(1:length(t)))
title('The Demodulator Output')
xlabel('Time')
```

解说题

解说题 3.8 [SSB-AM 解调中相位误差的后果]

在 SSB-AM 解调中,相位误差的后果是什么?

题 解

假定本地振荡器产生的正弦信号与载波有相位偏差 ϕ,则有:

$$\begin{aligned} y(t) &= u(t)\cos(2\pi f_c t + \phi) \\ &= \left[\frac{A_c}{2}m(t)\cos(2\pi f_c t) \mp \frac{A_c}{2}\hat{m}(t)\sin(2\pi f_c t)\right]\cos(2\pi f_c t + \phi) \\ &= \frac{A_c}{4}m(t)\cos\phi \pm \frac{A_c}{4}\hat{m}(t)\sin\phi + \text{高频项} \end{aligned} \tag{3.3.8}$$

可见,与 DSB-AM 不同,相位偏差的后果在这里不仅使解调信号衰减,而是解调信号既受到 $\cos\phi$ 因子的衰减,同时还附加了 $\pm(A_c/4)\hat{m}(t)\sin\phi$ 的失真项。在 $\phi=\pi/2$ 的特殊情况下,解调出了信号的希尔伯特变换,而不是解调了信号本身。

3.3.3 常规AM解调

我们已经看到，当考虑功率和SNR时，常规AM是比较差的调制方式，其原因是已调信号的大部分功率在不携带信息的载波分量上。载波分量的作用是通过包络检波器而使常规AM的解调更容易些；与此相反，对DSB-AM和SSB-AM则要求相干解调。因此，AM信号的解调比DSB-AM和SSB-AM信号的解调明显简单得多。所以，这种调制方法广泛用于广播系统中，在那里只有一台发射机，而有众多的接收机，接收机的价格应保持低廉。在包络检波中，经由二极管、电阻和电容器所组成的简单电路如图3.17所示，检测出已调信号的包络。

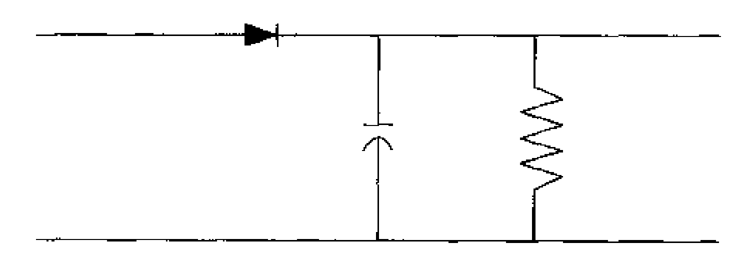


图3.17 一个简单的包络检波器

从数学上说，包络检波器产生常规AM信号的包络为：

$$V(t) = |1 + am_n(t)| \tag{3.3.9}$$

由于 $1 + m_n(t) \geqslant 0$，可以得到：

$$V(t) = 1 + am_n(t) \tag{3.3.10}$$

其中 $m_n(t)$ 正比于消息信号 $m(t)$，而1相应于载波分量，这个分量可用一个隔直流电路分隔开。正如前面所讨论的，它不需要知道载波相位的任何知识。这就是把这种解调方法称为**非相干解调**或**非同步解调**的缘故。回顾第1章中讨论到的，一个带通信号的包络可以表示成它的低通等效信号的幅度。因此，如果 $u(t)$ 是一个带通信号，其载波为 f_c，$u(t)$ 的低通等效信号记为 $u_l(t)$，那么 $u(t)$ 的包络 $V(t)$ 可以表示为：

$$V(t) = \sqrt{u_{lr}^2(t) + u_{li}^2(t)} = \sqrt{u_c^2(t) + u_s^2(t)} \tag{3.3.11}$$

这里的 $u_c(t)$ 和 $u_s(t)$ 代表带通信号 $u(t)$ 的同相和正交分量。因此，为了得到这个包络，只要求得该带通信号的低通等效信号就足够了，这个包络就是带通信号的低通等效的幅度。

解说题

解说题3.9 [包络检波]

有消息信号为：

$$m(t) = \begin{cases} 1, & 0 \leqslant t < \dfrac{t_0}{3} \\ -2, & \dfrac{t_0}{3} \leqslant t < \dfrac{2t_0}{3} \\ 0, & \text{其余 } t \end{cases}$$

用它以常规AM方式调制载波 $c(t) = \cos(2\pi f_c t)$，假设 $f_c = 250$ Hz，$t_0 = 0.15$ s，调制指数 $a = 0.85$。

1. 用包络检波器解调该信号,而载波分量又用隔直流电路阻隔掉,那么原消息 $m(t)$就被恢复了。注意,恢复 $m(t)$时的关键问题是,在所有 t 值中 $1+am_n(t)$都是正值,因此信号$[1+am_n(t)]\cos(2\pi f_c t)$的包络 $V(t)=|1+am_n(t)|$就等于 $1+am_n(t)$,从中很容易恢复出 $m(t)$。图 3.18 显示的是该常规 AM 已调信号和由包络检波器检测出的包络。在包络检波器分开已调信号的包络之后,信号的直流分量被去掉,并将信号幅度加权就得到了解调器输出。图 3.19 显示的是原消息信号和解调器输出。

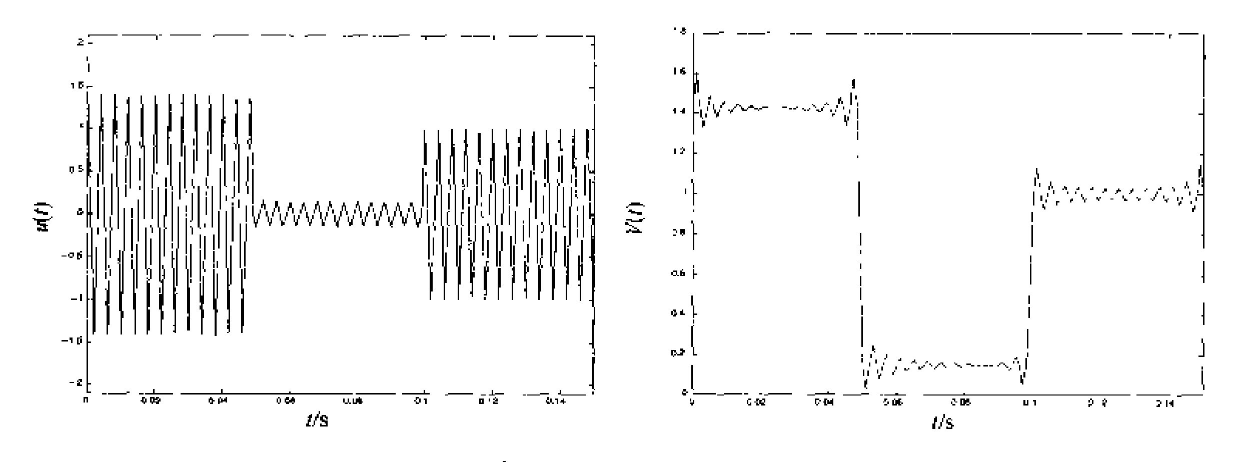


图 3.18 常规 AM 已调信号和它的包络

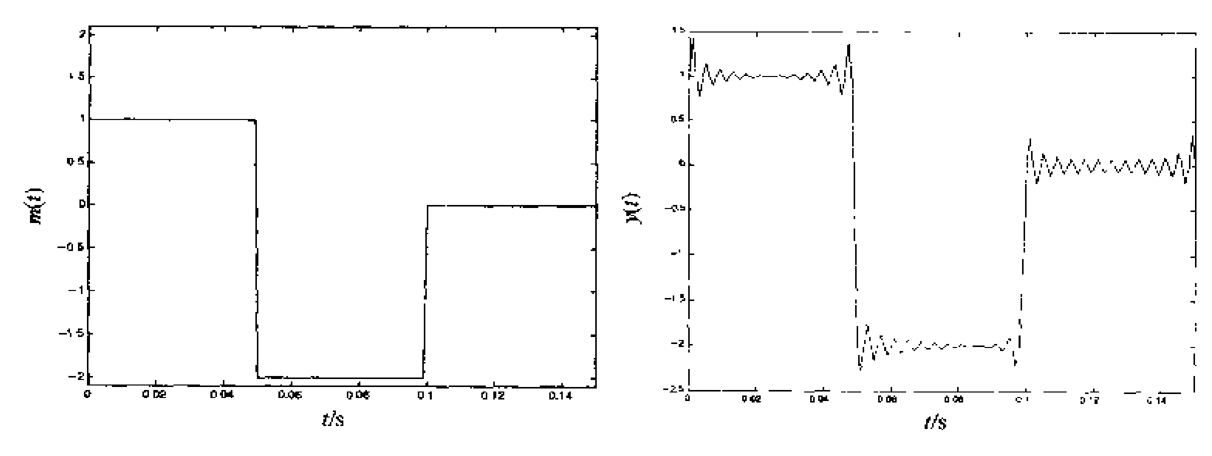


图 3.19 当无噪声存在时的消息信号和已解调信号

2. 当有噪声时,由于噪声的存在,信号会有一点失真。图 3.20 是已接收信号和它的包络,图 3.21 则对这种情况的消息信号和已解调信号进行了比较。

习 题

1. 与解说题 3.3 相同,有

$$u(t)=\left[1+0.85\,\frac{m(t)}{2}\right]\cos(2\pi f_c t)$$

$$=\left[1+0.425\Pi\left(\frac{t-0.025}{0.05}\right)-0.85\Pi\left(\frac{t-0.075}{0.05}\right)\right]\cos(500\pi t)$$

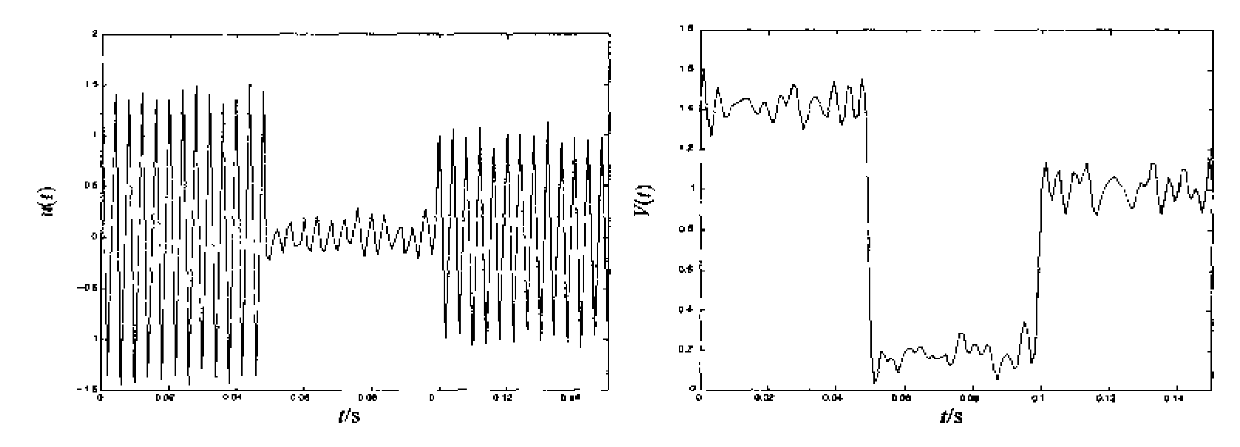


图 3.20　当有噪声存在时的已接收信号和它的包络

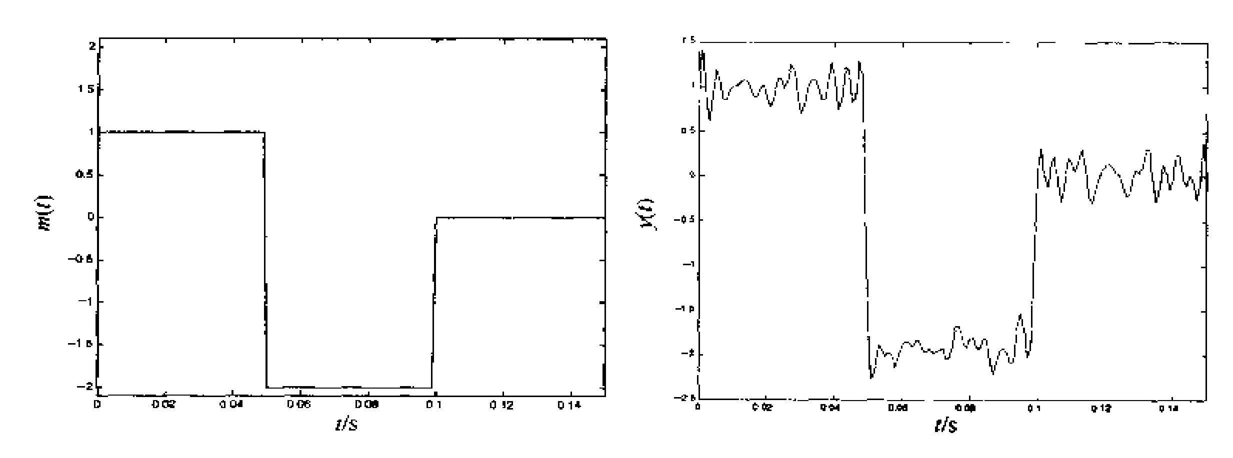


图 3.21　消息信号和有噪声存在时的已解调信号

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 3.9.
% Demonstration script for envelope detection. The message signal
% is +1 for 0 < t < t0/3, -2 for t0/3 < t < 2t0/3, and zero otherwise.
echo on
t0=.15;                                   % signal duration
ts=0.001;                                 % sampling interval
fc=250;                                   % carrier frequency
a=0.85;                                   % modulation index
fs=1/ts;                                  % sampling frequency
t=[0:ts:t0];                              % time vector
df=0.25;                                  % required frequency resolut
% message signal
m=[ones(1,t0/(3*ts)),-2*ones(1,t0/(3*ts)),zeros(1,t0/(3*ts)+1)];
c=cos(2*pi*fc.*t);                        % carrier signal
m_n=m/max(abs(m));                        % normalized message signa
[M,m,df1]=fftseq(m,ts,df);                % Fourier transform
f=[0:df1:df1*(length(m)-1)]-fs/2;         % frequency vector
u=(1+a*m_n).*c;                           % modulated signal
[U,u,df1]=fftseq(u,ts,df);                % Fourier transform
env=env_phas(u);                          % Find the envelope.
```

```
dem1=2*(env-1)/a;                          % Remove dc and rescale.
signal_power=spower(u(1:length(t)));       % power in modulated signal
noise_power=signal_power/100;              % noise power
noise_std=sqrt(noise_power);               % noise standard deviation
noise=noise_std*randn(1,length(u));        % Generate noise
r=u+noise;                                 % Add noise to the modulated signal.
[R,r,df1]=fftseq(r,ts,df);                 % Fourier transform
env_r=env_phas(r);                         % envelope, when noise is present
dem2=2*(env_r-1)/a;                        % Demodulate in the presence of noise.
pause  % Press any key to see a plot of the message.
subplot(2,1,1)
plot(t,m(1:length(t)))
axis([0 0.15 -2.1 2.1])
xlabel('Time')
title('The message signal')
pause  % Press any key to see a plot of the modulated signal.
subplot(2,1,2)
plot(t,u(1:length(t)))
axis([0 0.15 -2.1 2.1])
xlabel('Time')
title('The modulated signal')
pause  % Press a key to see the envelope of the modulated signal.
clf
subplot(2,1,1)
plot(t,u(1:length(t)))
axis([0 0.15 -2.1 2.1])
xlabel('Time')
title('The modulated signal')
subplot(2,1,2)
plot(t,env(1:length(t)))
xlabel('Time')
title('Envelope of the modulated signal')
pause  % Press a key to compare the message and the demodulated signal.
clf
subplot(2,1,1)
plot(t,m(1:length(t)))
axis([0 0.15 -2.1 2.1])
xlabel('Time')
title('The message signal')
subplot(2,1,2)
plot(t,dem1(1:length(t)))
xlabel('Time')
title('The demodulated signal')
pause  % Press a key to compare in the presence of noise.
clf
subplot(2,1,1)
plot(t,m(1:length(t)))
axis([0 0.15 -2.1 2.1])
xlabel('Time')
title('The message signal')
subplot(2,1,2)
plot(t,dem2(1:length(t)))
xlabel('Time')
title('The demodulated signal in the presence of noise')
```

注 释

在以上的解调过程中，已经略去了限制噪声滤波器的影响。这个滤波器是一个带通滤波器，它位于任何接收机的第一级。实际上，已接收的信号 $r(t)$ 先通过这个限制噪声滤波器，然

后再供给包络检波器。在前面的例子中，由于消息信号的带宽不是有限的，所以 $r(t)$ 通过任何带通滤波器都会引起在已解调消息上的失真，但是也会降低在解调器输出上的噪声量。图 3.22 已经画出了当利用不同带宽的限制噪声滤波器时解调器的输出，其中无限大带宽的情况就等效于图 3.21 的结果。

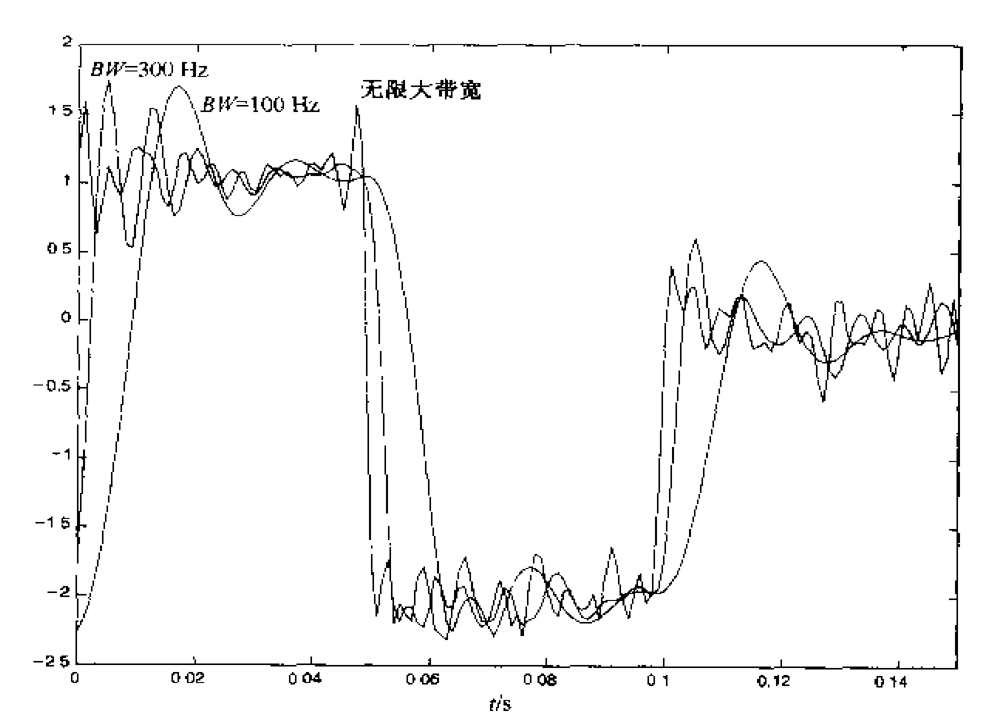


图 3.22　限制噪声滤波器的带宽对包络检波器输出的影响

3.4　角调制

角调制属于一类非线性调制方法，其中包括频率调制（FM）和相位调制（PM）。这类调制方法以其较高的带宽要求和在噪声存在情况下具有好的性能为特征。这些方法可以看成是以带宽来交换功率的调制技术，因此应用在带宽不是一个主要问题而希望有高的信噪比（SNR）的场合。频率调制广泛用于高保真度的 FM 广播、TV 音频广播、微波载波调制和点对点通信系统中。

在讨论角调制方法时，还是重点关注 5 个基本特性，包括时域表示、频域表示、带宽、功率和 SNR。因为 PM 和 FM 之间存在很密切的关系，所以将并行地讨论它们，但重点放在 FM 上。

当载波是 $c(t) = A_c\cos(2\pi f_c t)$，并且消息信号是 $m(t)$ 时，角调制信号的时域表示为：

$$u(t) = \begin{cases} A_c\cos(2\pi f_c t + k_p m(t)), & \text{PM} \\ A_c\cos\left(2\pi f_c t + 2\pi k_f\int_{-\infty}^{t} m(\tau)\mathrm{d}\tau\right), & \text{FM} \end{cases} \tag{3.4.1}$$

其中 k_f 和 k_p 分别表示 FM 和 PM 的**偏离常数**（deviation constant）。由于这些调制方法的非线性，一般来说角调制的频域表示很复杂，现仅讨论消息信号 $m(t)$ 是正弦信号的情况。假设对 PM 来说 $m(t) = a\cos(2\pi f_m t)$，对 FM 来说 $m(t) = -a\sin(2\pi f_m t)$。那么已调信号就具有如下

表示：

$$u(t)=\begin{cases}A_c\cos(2\pi f_c t+\beta_p\cos(2\pi f_m t)), & \text{PM}\\ A_c\cos(2\pi f_c t+\beta_f\cos(2\pi f_m t)), & \text{FM}\end{cases} \tag{3.4.2}$$

其中

$$\begin{cases}\beta_p=k_p a\\ \beta_f=\dfrac{k_f a}{f_m}\end{cases} \tag{3.4.3}$$

β_p 和 β_f 分别是PM和FM的**调制指数**。一般情况下，对于一个非正弦的 $m(t)$，调制指数定义为：

$$\begin{cases}\beta_p=k_p\max|m(t)|\\ \beta_f=\dfrac{k_f\max|m(t)|}{W}\end{cases} \tag{3.4.4}$$

其中 W 是消息信号 $m(t)$ 的带宽。在正弦消息信号中，已调信号可以表示为：

$$u(t)=\sum_{n=-\infty}^{\infty}A_c\mathrm{J}_n(\beta)\cos(2\pi(f_c+nf_m)t) \tag{3.4.5}$$

其中 $\mathrm{J}_n(\beta)$ 是 n 阶的第1类贝塞尔函数，而 β 是 β_p 或 β_f 之一，这取决于处理的是PM还是FM。在频域有：

$$U(f)=\sum_{n=-\infty}^{\infty}\left[\frac{A_c\mathrm{J}_n(\beta)}{2}\delta(f-(f_c+nf_m))+\frac{A_c\mathrm{J}_n(\beta)}{2}\delta(f+(f_c+nf_m))\right] \tag{3.4.6}$$

显然，这个已调信号的带宽不是有限的。然而，可以把包含已调信号功率的98%～99%的带宽定义为信号的**有效带宽**。这个带宽用**卡尔松法则**给出为：

$$B_T=2(\beta+1)W \tag{3.4.7}$$

其中，β 是调制指数，W 是消息信号的带宽，而 B_T 是已调信号的带宽。

角调制信号功率的表达式很简单。因为角调制信号是正弦的，具有变化的瞬时频率和恒定的幅度，它的功率是常数，与消息信号无关。无论是FM还是PM，其功率均为：

$$P_u=\frac{A_c^2}{2} \tag{3.4.8}$$

当不采用任何预加重和去加重过滤时，角调制信号的SNR由下式给出：

$$\left(\frac{S}{N}\right)_o=\begin{cases}\dfrac{P_M\beta_p^2}{(\max|m(t)|)^2}\dfrac{P_R}{N_0W}, & \text{PM}\\ 3\dfrac{P_M\beta_f^2}{(\max|m(t)|)^2}\dfrac{P_R}{N_0W}, & \text{FM}\end{cases} \tag{3.4.9}$$

因为消息信号的最大幅度记为 $\max|m(t)|$，所以可将 $P_M/(\max|m(t)|)^2$ 理解为**归一化消息信号的功率**，并记为 P_{M_n}。当使用了3 dB的截止频率为 f_0 的预加重和去加重滤波器时，对FM的SNR给出为：

$$\left(\frac{S}{N}\right)_{oPD}=\frac{(W/f_0)^3}{3[W/f_0-\arctan(W/f_0)]}\left(\frac{S}{N}\right)_o \tag{3.4.10}$$

其中 $(S/N)_o$ 是由式(3.4.9)给出的无预加重和去加重过滤时的SNR。

解说题

解说题 3.10　[频率调制]

用消息信号

$$m(t)=\begin{cases}1, & 0\leqslant t<\frac{t_0}{3}\\ -2, & \frac{t_0}{3}\leqslant t<\frac{2t_0}{3}\\ 0, & 其余\ t\end{cases}$$

采用频率调制方法调制载波 $c(t)=\cos(2\pi f_c t)$，假设 $f_c=200$ Hz，$t_0=0.15$ s，偏离常数 $k_f=50$。

1. 画出已调信号。

2. 求消息信号和已调信号的频谱。

题　解

1. 现有

$$u(t)=A_c\cos\left(2\pi f_c t+2\pi k_f\int_{-\infty}^{t}m(\tau)\mathrm{d}\tau\right)$$

我们必须求 $\int_{-\infty}^{t}m(\tau)\mathrm{d}\tau$。这可以用数值法或解析法来完成，其结果如图 3.23 所示。利用 $u(t)$ 的关系和 $m(t)$ 的积分值，就可以求得 $u(t)$ 的表达式。$m(t)$ 和 $u(t)$ 的图见图 3.24。

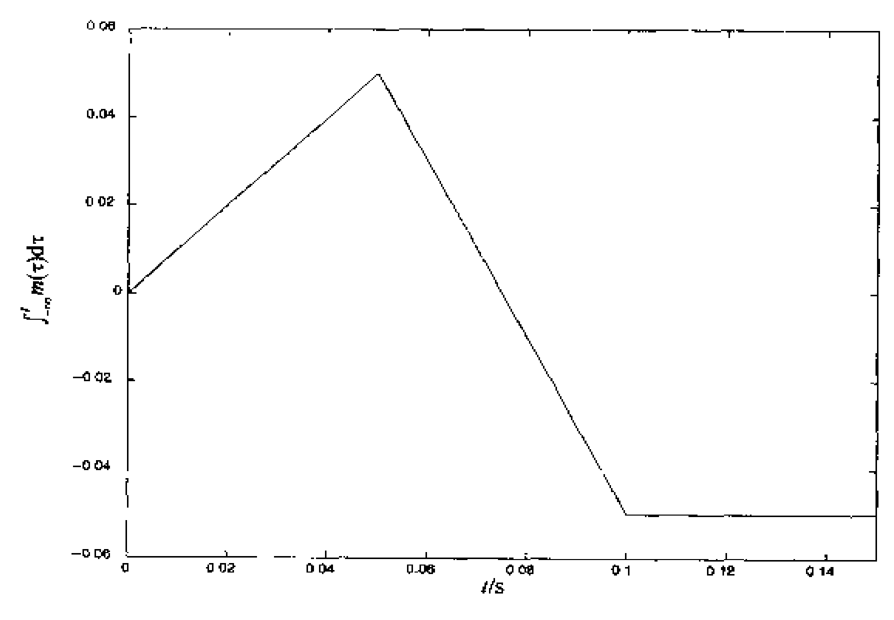


图 3.23　消息信号的积分

2. 利用 MATLAB 的傅里叶变换子程序，可得 $u(t)$ 的频谱，如图 3.25 所示。很容易看出，与 AM 不同，在 FM 情况下消息信号频谱和已调信号频谱之间不存在明显的相似性。在现在这个特例中，消息信号的带宽不是有限的，因此，为了定义调制指数，应该在表达式

$$\beta=\frac{k_f\max|m(t)|}{W}\tag{3.4.11}$$

中采用一个消息信号的近似带宽。例如,可以用消息信号 $m(t)$ 频谱的主瓣宽度作为带宽,这样可以得到:

$$W = 20\ \text{Hz}$$

所以

$$\beta = \frac{50 \times 2}{20} = 5$$

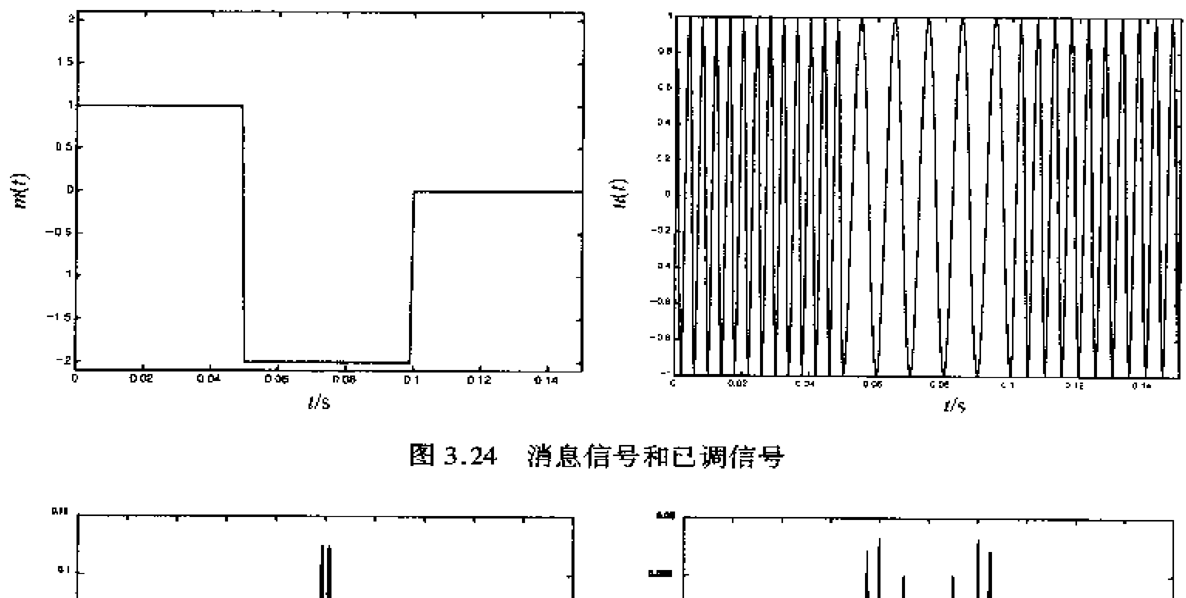
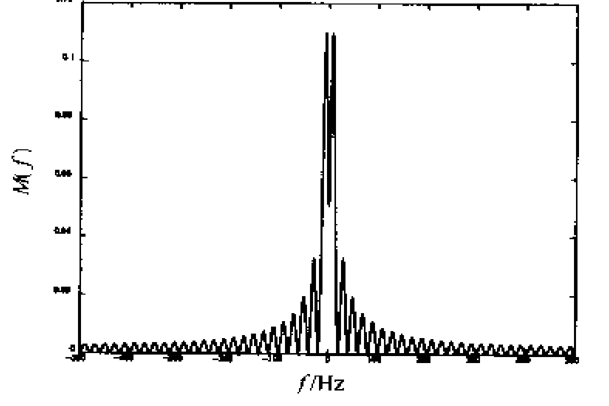


图 3.24 消息信号和已调信号



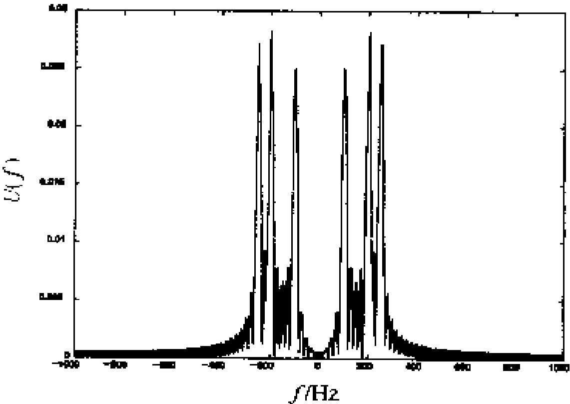


图 3.25 消息信号和已调信号的幅度谱

本题的 MATLAB 脚本如下所示。

m文件

```
% MATLAB script for Illustrative Problem 3.10.
% Demonstration script for frequency modulation. The message signal
% is +1 for 0 < t < t0/3, -2 for t0/3 < t < 2t0/3, and zero otherwise.
echo on
t0=.15;                          % signal duration
ts=0.0005;                       % sampling interval
fc=200;                          % carrier frequency
kf=50;                           % modulation index
fs=1/ts;                         % sampling frequency
t=[0:ts:t0];                     % time vector
df=0.25;                         % required frequency resolution
```

```
% message signal
m=[ones(1,t0/(3*ts)),-2*ones(1,t0/(3*ts)),zeros(1,t0/(3*ts)+1)];
int_m(1)=0;
for i=1:length(t)-1                       % integral of m
  int_m(i+1)=int_m(i)+m(i)*ts;
  echo off ;
end
echo on ;
[M,m,df1]=fftseq(m,ts,df);                % Fourier transform
M=M/fs;                                   % scaling
f=[0:df1:df1*(length(m)-1)]-fs/2;         % frequency vector
u=cos(2*pi*fc*t+2*pi*kf*int_m);           % modulated signal.
[U,u,df1]=fftseq(u,ts,df);                % Fourier transform
U=U/fs;                                   % scaling
pause  % Press any key to see a plot of the message and the modulated signal.
subplot(2,1,1)
plot(t,m(1:length(t)))
axis([0 0.15 -2.1 2.1])
xlabel('Time')
title('The message signal')
subplot(2,1,2)
plot(t,u(1:length(t)))
axis([0 0.15 -2.1 2.1])
xlabel('Time')
title('The modulated signal')
pause   % Press any key to see plots of the magnitude of the message and the
        % modulated signal in the frequency domain.
subplot(2,1,1)
plot(f,abs(fftshift(M)))
xlabel('Frequency')
title('Magnitude spectrum of the message signal')
subplot(2,1,2)
plot(f,abs(fftshift(U)))
title('Magnitude spectrum of the modulated signal')
xlabel('Frequency')
```

解说题

解说题 3.11　[频率调制]

设消息信号 $m(t)$ 为：

$$m(t)=\begin{cases}\text{sinc}(100t), & |t|\leqslant t_0\\ 0, & \text{其余 } t\end{cases}$$

其中 $t_0=0.1$。用这个消息信号调制载波 $c(t)=\cos(2\pi f_c t)$，$f_c=250$ Hz，偏离常数 $k_f=100$。

1. 在时域和频域中画出已调信号。
2. 将解调器输出与原消息信号进行比较。

题　解

1. 首先对消息信号积分，然后用下面的关系求出 $u(t)$。

$$u(t)=A_c\cos\left(2\pi f_c t+2\pi k_f\int_{-\infty}^{t}m(\tau)\mathrm{d}\tau\right)$$

$u(t)$和消息信号如图 3.26 所示。该消息信号的积分如图 3.27 所示，已调信号在频域中的频谱见图 3.28。

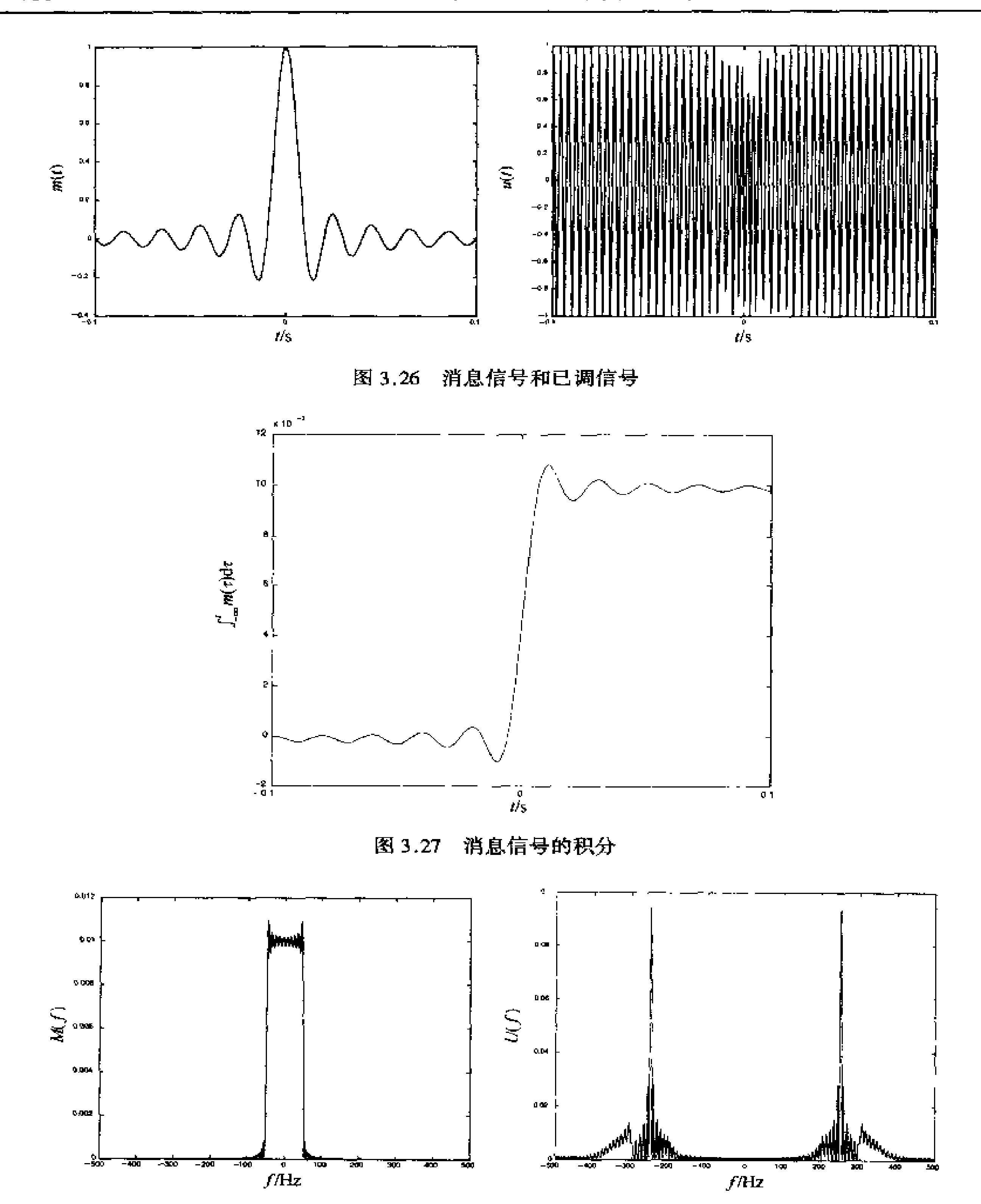


图 3.26 消息信号和已调信号

图 3.27 消息信号的积分

图 3.28 消息信号和已调信号的幅度谱

2. 为了对 FM 信号解调,首先要求出已调信号 $u(t)$ 的相位。这个相位是 $2\pi k_f\int_{-\infty}^{t} m(\tau)d\tau$,求微分并除以 $2\pi k_f$ 就得到了 $m(t)$。注意,为了恢复这个相位,并将 2π 的相位卷绕解开,使用了

MATLAB 的 unwrap.m 函数。图 3.29 显示的是该消息信号和已解调信号的图。正如读者所看到的，这个已解调信号非常类似于消息信号。

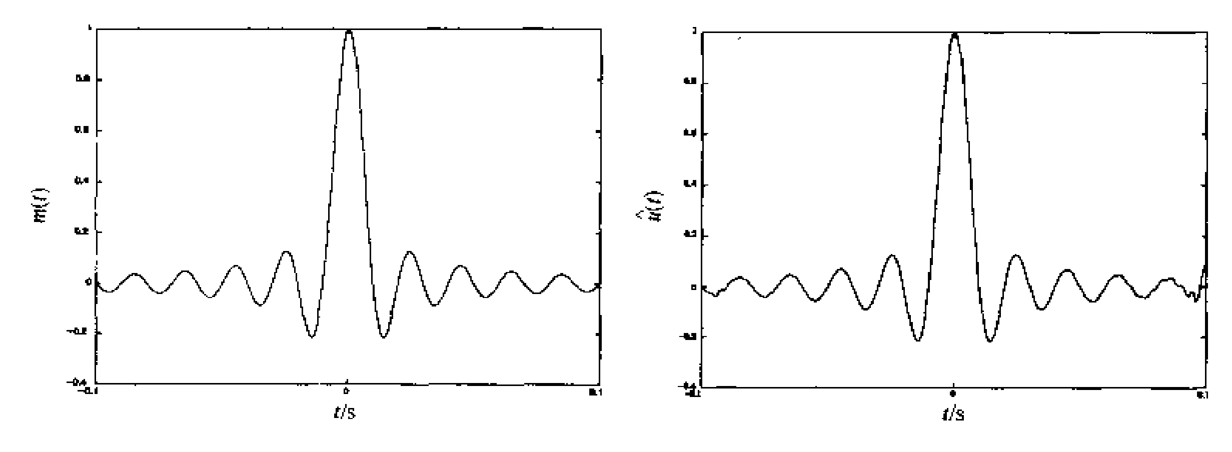


图 3.29 消息信号和已解调信号

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 3.11.
% Demonstration script for frequency modulation. The message signal
% is m(t)=sinc(100t).
echo on
t0=.2;                                   % signal duration
ts=0.001;                                % sampling interval
fc=250;                                  % carrier frequency
snr=20;                                  % SNR in dB (logarithmic)
fs=1/ts;                                 % sampling frequency
df=0.3;                                  % required freq. resolution
t=[-t0/2:ts:t0/2];                       % time vector
kf=100;                                  % deviation constant
df=0.25;                                 % required frequency resolution
m=sinc(100*t);                           % the message signal
int_m(1)=0;
for i=1:length(t)-1                      % integral of m
  int_m(i+1)=int_m(i)+m(i)*ts;
  echo off ;
end
echo on ;
[M,m,df1]=fftseq(m,ts,df);               % Fourier transform
M=M/fs;                                  % scaling
f=[0:df1:df1*(length(m)-1)]-fs/2;        % frequency vector
u=cos(2*pi*fc*t+2*pi*kf*int_m);          % modulated signal
[U,u,df1]=fftseq(u,ts,df);               % Fourier transform
U=U/fs;                                  % scaling
[v,phase]=env_phas(u,ts,250);            % demodulation, find phase of u
phi=unwrap(phase);                       % Restore original phase.
dem=(1/(2*pi*kf))*(diff(phi)/ts);        % demodulator output, differentiate and scale phase
pause % Press any key to see a plot of the message and the modulated signal.
subplot(2,1,1)
plot(t,m(1:length(t)))
xlabel('Time')
title('The message signal')
```

```
subplot(2,1,2)
plot(t,u(1:length(t)))
xlabel('Time')
title('The modulated signal')
pause    % Press any key to see plots of the magnitude of the message and the
         % modulated signal in the frequency domain.
subplot(2,1,1)
plot(f,abs(fftshift(M)))
xlabel('Frequency')
title('Magnitude spectrum of the message signal')
subplot(2,1,2)
plot(f,abs(fftshift(U)))
title('Magnitude-spectrum of the modulated signal')
xlabel('Frequency')
pause    % Press any key to see plots of the message and the demodulator output with no
         % noise.
subplot(2,1,1)
plot(t,m(1:length(t)))
xlabel('Time')
title('The message signal')
subplot(2,1,2)
plot(t,dem(1:length(t)))
xlabel('Time')
title('The demodulated signal')
```

思考题

一个频率调制信号有恒定的幅度,然而在图3.26中,信号 $u(t)$ 的幅度明显不是恒定的。能解释为什么会这样吗?

3.5 习题

3.1 信号 $m(t)$ 在区间[0,2]内给出为:

$$m(t)=\begin{cases} t, & 0.1\leqslant t<1 \\ -t+2, & 1\leqslant t<1.9 \\ 0.1, & \text{其余 } t \end{cases}$$

使用该信号以DSB方式调制一个载波频率为25 Hz,幅度为1的载波,产生已调信号 $u(t)$。写一个MATLAB m文件,并用该文件做下面的习题:

a. 画出已调信号;

b. 求已调信号的功率;

c. 求已调信号的频谱;

d. 求已调信号的功率谱密度并与消息信号的功率谱密度进行比较。

3.2 信号 $m(t)$ 在区间[0,2]内给出为:

$$m(t)=\begin{cases} t, & 0\leqslant t<1 \\ -t+2, & 1\leqslant t\leqslant 2 \end{cases}$$

用 $m(t)$ 重做习题3.1。本题和习题3.1之间的差别是什么?

3.3 信号 $m(t)$ 给出为:

$$m(t)=\begin{cases}\operatorname{sinc}^2(10t), & |t|\leqslant 2\\ 0, & \text{其余 } t\end{cases}$$

用 $m(t)$和频率为 100 Hz 的载波重做习题 3.1。

3.4　假设习题 3.1 中不采用 DSB 方式,而是采用常规 AM 方式,调制指数 $a=0.2$。

a. 求出并画出已调信号的频谱;

b. 将调制指数从 0.1 改变到 0.9,解释这样会如何影响由上面 a 部分导得的频谱;

c. 作为调制指数的函数,画出边带功率对载波功率的比率。

3.5　在习题 3.1 中,设调制方式是 USSB 而不是 DSB。首先用 DSB 方式,然后再除去下边带。

a. 求出并画出已调信号;

b. 求出并画出已调信号的频谱;

c. 将已调信号的频谱与未调制信号的频谱进行比较。

3.6　在习题 3.5 中,不用滤波 DSB 产生 USSB 信号,而用下面的关系:

$$u(t)=\frac{A_c}{2}m(t)\cos(2\pi f_c t)-\frac{A_c}{2}\hat{m}(t)\sin(2\pi f_c t)$$

能看出与习题 3.5 得到的结果有什么差别吗?

3.7　改用消息信号 $m(t)$:

$$m(t)=\begin{cases}\operatorname{sinc}^2(10t), & |t|\leqslant 2\\ 0, & \text{其余 } t\end{cases}$$

并用频率为 100 Hz 的载波产生 LSSB 信号,重做习题 3.5 和习题 3.6。

3.8　用信号

$$m(t)=\begin{cases}t, & 0.1\leqslant t<1\\ -t+2, & 1\leqslant t<1.9\\ 0.1, & \text{其余 } t\end{cases}$$

以 DSB 调制方式调制频率为 25 Hz 的载波。

a. 求出并画出已调信号;

b. 假设在解调器端的本地振荡器与载波有相位为 θ 的滞后,$0\leqslant\theta\leqslant\pi/2$,利用 MATLAB 画出已调信号的功率与 θ 的依赖关系图(假定无噪声传输)。

3.9　在习题 3.8 中,假设调制方式是 USSB,画出在 $\theta=0°,30°,45°,60°$和 $90°$时的已解调信号。

3.10　假设调制是调制指数为 0.2 的常规 AM,重做习题 3.8。并用包络检波器解调该信号,当无噪声存在时,画出已解调信号和它的频谱。

3.11　消息信号 $m(t)$是周期的,周期为 2 s,在区间[0,2]内定义为:

$$m(t)=\begin{cases}t, & 0.1\leqslant t<1\\ -t+2, & 1\leqslant t<1.9\\ 0, & \text{其余 } t\end{cases}$$

用该信号以 DSB 方式调制频率为 50 Hz 的载波。画出这个 DSB 解调器的输出,并将它与在下述噪声情况下的消息信号进行比较:高斯白噪声的功率是已调信号功率的 0.001,0.01,0.05,0.1 和 0.3 倍,并加在已调信号上。

3.12　用 LSSB 方式重做习题 3.11。将结果与习题 3.11 的结果进行比较。

3.13　用常规 AM 和包络解调重做习题 3.11。

3.14 用信号

$$m(t)=\begin{cases}t, & 0\leqslant t<1\\ -t+2, & 1\leqslant t\leqslant 2\\ 0.1, & 其余\ t\end{cases}$$

对频率为 1000 Hz 的载波进行频率调制 FM,偏离常数 $k_f=25$。

a. 求已调信号的瞬时频率范围;

b. 求已调信号的带宽;

c. 画出消息信号和已调信号的频谱;

d. 求调制指数。

3.15 利用频率解调的 MATLAB 文件对习题 3.14 的已调信号进行解调,并将已解调信号与消息信号进行比较。

3.16 设消息信号是周期的,周期为 2,在区间[0,2]内定义为:

$$m(t)=\begin{cases}t, & 0.1\leqslant t<1\\ -t+2, & 1\leqslant t<1.9\\ 0.1, & 其余\ t\end{cases}$$

调制方式与习题 3.14 的相同。在解调以前,将加性高斯白噪声加到已调信号上。当噪声功率与已调信号功率之比是 0.001,0.01,0.05,0.1 和 0.3 时,解调并画出已解调信号。

3.17 在习题 3.16 中,设 $m(t)=0$,按题中所述将噪声过程加到已调信号上,并解调出所得的信号,画出在每种情况下解调器输出的功率谱密度。

第4章　模拟－数字转换

4.1　概述

大多数信息源原本都是模拟的，模拟(信息)源包括语音、图像以及许多遥测源。本章要讨论用有效的方式将模拟源转换成数字序列的各种方法和技术。在后续各章中将会看到，由于数字信息更容易处理、通信和存储，将模拟源转换成数字序列是很有必要的。**数据压缩**的一般论题(其中，模拟－数字转换是一种特例)可以分为两个主要分支：

1. **量化**(或称**有损数据压缩**)，其中模拟源被量化到某个有限电平数。在这个过程中，产生失真是不可避免的，所以会丢掉某些信号，而且这个丢失了的信息不可能恢复。一般的模拟－数字转换技术，例如脉冲编码调制(PCN)、差分脉冲编码调制(DPCM)、Δ调制(ΔM)、均匀量化、非均匀量化和矢量量化等都属于这一类。这类数据压缩方法在性能上的基本限制是由**率失真界**(rate-distortion bound)给出的。

2. **无噪声编码**(或称**无损数据压缩**)，其中数字数据(通常是上面讨论的量化结果)被压缩，以达到用尽可能少的比特数来表示它们，这样使得原数据序列能够完全从已压缩的序列中恢复出来。信源编码技术，如 Huffman 编码，Lempel-Ziv 编码以及算术编码等都属于这一类数据压缩方法。在这类编码方法中，没有任何信息丢失。这类压缩方法所实现的压缩的基本限制由信源的**熵**给出。

4.2　信息的度量

一个信号源(数据、语音、视频影像等)的输出能够用一个随机过程来建模。对一个离散无记忆和平稳的随机过程(可以认为它是一个随机变量 X 的独立抽取)来说，信息含量(或熵)定义为：

$$H(X) = -\sum_{x\in \mathscr{X}} p(x)\log_2 p(x) \tag{4.2.1}$$

其中，$\mathscr{X}$ 表示信源字符集，$p(x)$是字符 x 的概率。这样得出的熵以比特表示。对于概率为 p 和 $1-p$ 的二进制字符来说，熵用 $H_b(p)$表示并给出为：

$$H_b(p) = -p\log_2 p - (1-p)\log_2(1-p) \tag{4.2.2}$$

图4.1给出的是二进制熵函数的图。

一个信源的熵给出了完全恢复这个信源所要求的表示这个信源的比特数的基本界限。换句话说，为了进行无误差的恢复，信源编码所要求的每个信源输出的平均比特数可以实现到尽量接近 $H(X)$，但决不能小于 $H(X)$。

4.2.1　无噪声编码

无噪声编码是所有这类方法的一般称呼，这类方法在达到完全恢复的前提下，降低了某个

信源输出表示所要求的比特数。无噪声编码理论(归功于 Shannon)指出,有可能使用一种码,它的码率可以尽量接近这个信源的熵,以完全恢复这个信源,但小于信源熵的码率的码不可能完全恢复这个信源。换句话说,无论编码器和解码器的复杂性如何,对任意 $\varepsilon>0$,可以有一种码,其码率小于 $H(X)+\varepsilon$,而不能有一种码率小于 $H(X)$。现在有各种无噪声信源编码算法,Huffman 编码和 Lempel-Ziv 编码就是其中的两个例子。现在我们来讨论 Huffman 编码算法。

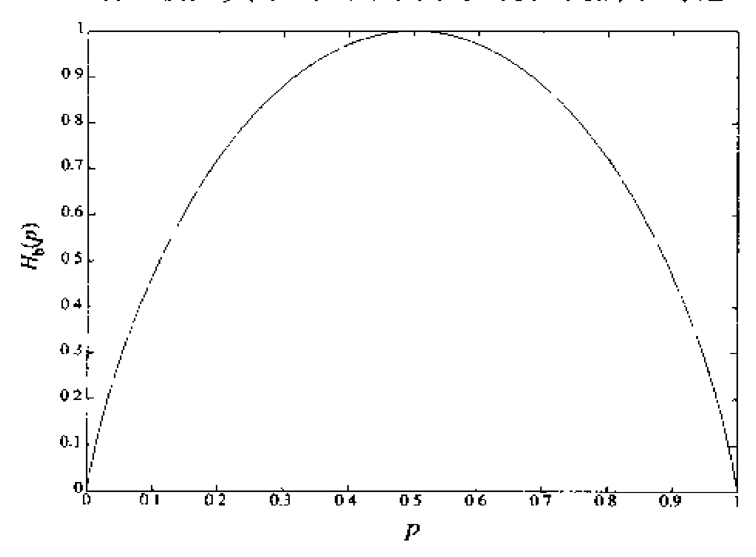


图 4.1 二进制熵函数的图

Huffman 编码

在 Huffman 编码中,将较长的码字分配给较小可能的信源输出,而将较短的码字分配给较大可能的信源输出。为了实现这一点,先将两个最低概率的信源输出合并,产生一个新的合并后的输出,它的概率是两个相应概率的和。将这一过程重复下去,直到仅有一个合并了的输出为止。这样就得到了一个树状的图。然后,从树的根部开始,将 0 和 1 放到合并成同一节点的任意两条支路上,这样就产生了这组码。可以证明,用这种方式所产生的码在**异前缀码**(prefix-free code)[①] 中具有最小平均长度。下面的例子说明了如何来设计一个 Huffman 码。

解说题

解说题 4.1 [Huffman 码]

为某个信源设计一个 Huffman 码,该信源的字符集为 $\mathcal{X}=\{x_1,x_2,\cdots,x_9\}$,相应的概率向量为:

$$\boldsymbol{p}=(0.2,0.15,0.13,0.12,0.1,0.09,0.08,0.07,0.06)$$

求所得码的平均码长,并与该信源的熵进行比较。

题 解

依据上面所提出的算法得到的树如图 4.2 所示。这个码的平均码字长度是:

$$\bar{L}=2\times0.2+3\times(0.15+0.13+0.12+0.1)+4\times(0.09+0.08+0.07+0.06)$$

① 异前缀码是这样一类码,其中没有一个码字是另一个码字的前缀。

=3.1　比特/信源输出

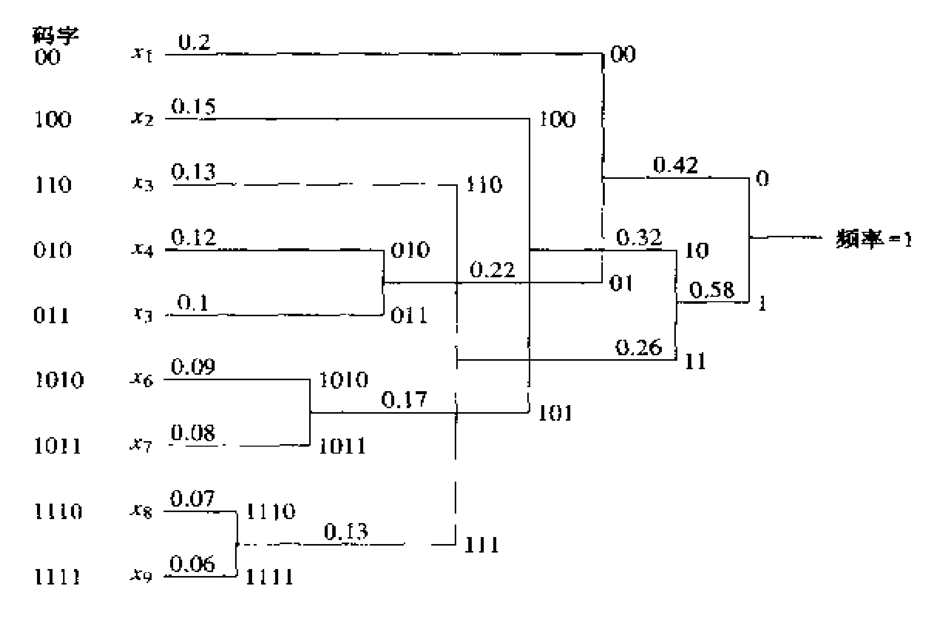


图 4.2　Huffman 码树

信源的熵为：

$$H(X) = -\sum_{i=1}^{9} p_i \log_2 p_i = 3.0371 \quad \text{比特 / 信源输出}$$

可见，正如所预期的，$\bar{L} > H(X)$。

下面给出的 MATAB 函数 entropy.m 可计算出概率向量 $\boldsymbol{p}$ 的熵。

m 文件

```
function h=entropy(p)
%               H=ENTROPY(P) returns the entropy function of
%               the probability vector p.
if length(find(p<0))~=0,
  error('Not a prob. vector, negative component(s)')
end
if abs(sum(p)-1)>10e-10,
  error('Not a prob. vector, components do not add up to 1')
end
h=sum(-p.*log2(p));
```

下面这个量：

$$\eta = \frac{H(X)}{\bar{L}} \tag{4.2.3}$$

称为 **Huffman 码的效率**，虽然总是有 $\eta \leqslant 1$。一般来说，可以证明任何 Huffman 码的平均码字长度都满足下面的不等式

$$H(x) \leqslant \bar{L} < H(X) + 1 \tag{4.2.4}$$

如果是对长度为 K 的分组码设计一个 Huffman 码，而不是对单个字符进行设计，就有：

$$H(X) \leqslant \bar{L} < H(X) + \frac{1}{K} \tag{4.2.5}$$

因此,增大 K 就能随意逼近 $H(X)$。毫无疑问,增大 K 会大大增加复杂性。值得注意的是,Huffman 码算法不会得出惟一的码,这是由于在不同的树支路处可任意置 0 和 1。这就是我们总说"某个"Huffman 码,而不说"该"Huffman 码的缘故。

下面给出的 MATLAB 函数 huffman.m 为概率向量 p 的离散无记忆信源设计某个 Huffman 码,并能得到码字和平均码字长度。

m 文件

```
function [h,l]=huffman(p);
%HUFFMAN        Huffman code generator
%               [h,l]=huffman(p), Huffman code generator
%               returns h the Huffman code matrix, and l the
%               average codeword length for a source with
%               probability vector p.

if length(find(p<0))~=0,
  error('Not a prob. vector, negative component(s)')
end
if abs(sum(p)-1)>10e-10,
  error('Not a prob. vector, components do not add up to 1')
end
n=length(p);
q=p;
m=zeros(n-1,n);
for i=1:n-1
  [q,l]=sort(q);
  m(i,:)=[l(1:n-i+1),zeros(1,i-1)];
  q=[q(1)+q(2),q(3:n),1];
end
for i=1:n-1
  c(i,:)=blanks(n*n);
end
c(n-1,n)='0';
c(n-1,2*n)='1';
for i=2:n-1
  c(n-i,1:n-1)=c(n-i+1,n*(find(m(n-i+1,:)==1))...
  -(n-2):n*(find(m(n-i+1,:)==1)));
  c(n-i,n)='0';
  c(n-i,n+1:2*n-1)=c(n-i,1:n-1);
  c(n-i,2*n)='1';
  for j=1:i-1
    c(n-i,(j+1)*n+1:(j+2)*n)=c(n-i+1,...
    n*(find(m(n-i+1,:)==j+1)-1)+1:n*find(m(n-i+1,:)==j+1));
  end
end
for i=1:n
  h(i,1:n)=c(1,n*(find(m(1,:)==i)-1)+1:find(m(1,:)==i)*n);
  l1(i)=length(find(abs(h(i,:))~=32));
end
l=sum(p.*l1);
```

解说题

解说题 4.2　[Huffman 编码]

有一个离散无记忆信息源,其字符集为:

$$\mathcal{X} = \{x_1, x_2, \cdots, x_6\}$$

对应的概率为

$$p = \{0.1, 0.3, 0.05, 0.09, 0.21, 0.25\}$$

对该信源用 Huffman 编码。

1. 求信源的熵。

2. 求该信源的 Huffman 码，并求 Huffman 码的效率。

3. 现在对长度为 2 的信源序列设计一个 Huffman 码，比较这个码的效率和第 2 步中得出的码的效率。

题 解

1. 经由函数 entropy.m 得出该信源的熵是 2.3549 比特/信源符号。

2. 利用 huffman.m 函数就能对该信源设计一个 Huffman 码，求得的码字是 010, 11, 0110, 0111, 00 和 10。平均码字长度是 2.38 比特/信源输出，因此该码的效率是：

$$\eta_1 = \frac{2.3549}{2.38} = 0.9895$$

3. 新信源的信源输出是原信源字符对，有 36 个输出字符，形式为$\{(x_i, x_j)\}_{i,j=1}^{6}$。因为这个信源是无记忆的，所以每对概率就是单个字符概率的乘积。因此，为了得到这个新信源的概率向量，必须产生有 36 个元素的向量，其中每个元素是原概率向量 p 中两个概率的乘积，这可以使用 MATLAB 函数 kron.m 以 kron(p,p)的形式来完成。这个 Huffman 码字如下所示：

1110000, 01110, 10110111, 1011001, 111001, 00101, 01111, 000, 011010, 00111, 1001, 1100, 11101110, 011011, 111011110, 111011111, 1110001, 001000, 1011010, 01100, 10110110, 1011000, 101110, 111110, 111010, 1010, 1110110, 101111, 11110, 0100, 00110, 1101, 001001, 111111, 0101, 1000

这个新信源的平均码字长度是 4.7420，而它的熵是 4.7097，所以这个 Huffman 码的效率是：

$$\eta_2 = \frac{4.7097}{4.7420} = 0.9932$$

与本题第 2 步设计出的 Huffman 码效率相比是有改善的。

解说题

解说题 4.3 [最大效率的 Huffman 码]

设计某个信源的 Huffman 码，其概率向量为：

$$p = \left(\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32}, \frac{1}{64}, \frac{1}{128}, \frac{1}{256}, \frac{1}{256}\right)$$

题 解

用 Huffman.m 函数求一个 Huffman 码和对应的平均码字长度。所得码字是 1, 01, 001, 0001, 00001, 000001, 0000001, 00000000 和 00000001，平均码字长度是 1.9922 比特/信源输出。如果用 entropy.m 函数求这个信源的熵，也可得 1.9922 比特/信源输出，所以这个码的效率是 1。

思考题

在什么条件下，一个 Huffman 码的效率等于 1？

4.3 量化

前一节讨论了两种无噪声编码方法,也就是说,将信源输出序列压缩,并使得从已压缩的数据中可以全部恢复出原信源。在这些方法中,已压缩的数据是信源输出的一个确定性函数,而信源输出也是已压缩数据的一个确定性函数。这种已压缩数据和信源输出之间的一对一的对应关系意味着它们的熵是相等的,在编码和解码过程中没有信息丢失。

在很多应用中(例如,模拟信号的数字处理),信源的字符集不是离散的,为表示每个信源输出所需的比特数就不是有限的了。为了用数字方式处理信源,不得不把信源**量化**到某个有限的数值上。这个过程虽然将比特数减少到某个有限的数上,但同时却引入了某些失真。在量化过程中丢失的信息是永远不可能再恢复的。

一般来说,量化方法可分为**标量量化**和**矢量量化**。在标量量化中每个信源输出单独量化,而在矢量量化中信源输出分组被量化。

可以将标量量化器进一步分为**均匀量化器**和**非均匀量化器**。在均匀量化中,将量化的区域选为相等长度;在非均匀量化中,允许有各种不同长度的区域。很显然,一般情况下非均匀量化器要优于均匀量化器。

4.3.1 标量量化

在标量量化中,随机变量 X 的范围被分成 N 个互不重叠的区域 $\mathcal{R}_i, 1 \leqslant i \leqslant N$,称为**量化间隔**,在每个区域内选取的某个单一的点称为**量化电平**。然后,随机变量落进区域 $\mathcal{R}_i$ 内的所有的值都被量化到 i 个量化电平上,用 $\hat{x}_i$ 表示,这就意味着:

$$x \in \mathcal{R}_i \Leftrightarrow Q(x) = \hat{x}_i \tag{4.3.1}$$

其中

$$\hat{x}_i \in \mathcal{R}_i \tag{4.3.2}$$

显然,这种形式的量化就引入了均方程差 $(x - \hat{x}_i)^2$,因此这个均方量化误差由下式给出:

$$D = \sum_{i=1}^{N} \int_{\mathcal{R}_i} (x - \hat{x}_i)^2 f_X(x) \mathrm{d}x \tag{4.3.3}$$

其中 $f_X(x)$ 代表信源随机变量的概率密度函数。**信号量化噪声比**(SQNR)定义为:

$$\mathrm{SQNR}|_{\mathrm{dB}} = 10 \log_{10} \frac{E[X^2]}{D}$$

均匀量化

在均匀量化中,除去第一个和最后一个区域(也就是 $\mathcal{R}_1$ 和 $\mathcal{R}_N$)以外,全部量化区域都具有相等的长度并记为 Δ,因此有:

$$\begin{aligned}
\mathcal{R}_1 &= (-\infty, a] \\
\mathcal{R}_2 &= (a, a+\Delta] \\
\mathcal{R}_3 &= (a+\Delta, a+2\Delta] \\
&\vdots \\
\mathcal{R}_N &= (a+(N-2)\Delta, \infty)
\end{aligned}$$

在每个量化间隔内,可以证明最佳的量化电平是这个间隔的**质心**(centroid),即

$$\hat{x}_i = E[X \mid X \in \mathscr{R}_i] = \frac{\int_{\mathscr{R}_i} x f_X(x)\mathrm{d}x}{\int_{\mathscr{R}_i} f_X(x)\mathrm{d}x} \qquad 1 \leqslant i \leqslant N \tag{4.3.4}$$

因此，均匀量化器的设计就等效于确定 a 和 Δ。在确定出 a 和 Δ 之后，$\hat{x}_i$ 的值和产生的失真很容易用式(4.3.3)和式(4.3.4)确定。在某些情况下，为了方便，就选量化区域的中点作为量化电平，也就是距离量化区域边界 $\Delta/2$ 处。

图 4.3 和图 4.4 分别显示的是 N 为偶数值和奇数值时某个对称概率密度函数的量化函数 $Q(x)$ 的图。

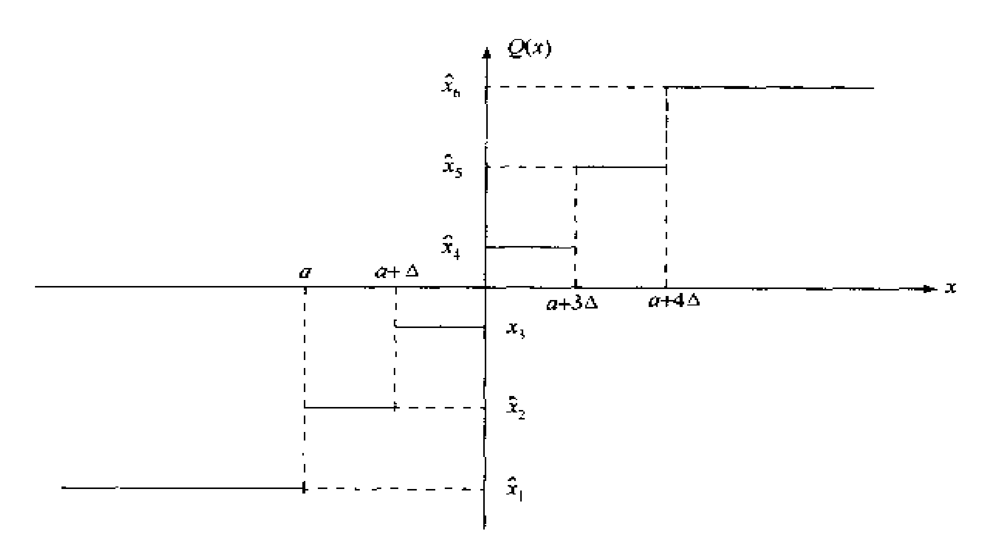


图 4.3　$N=6$ 的均匀量化器(注意这里 $a+2\Delta=0$)

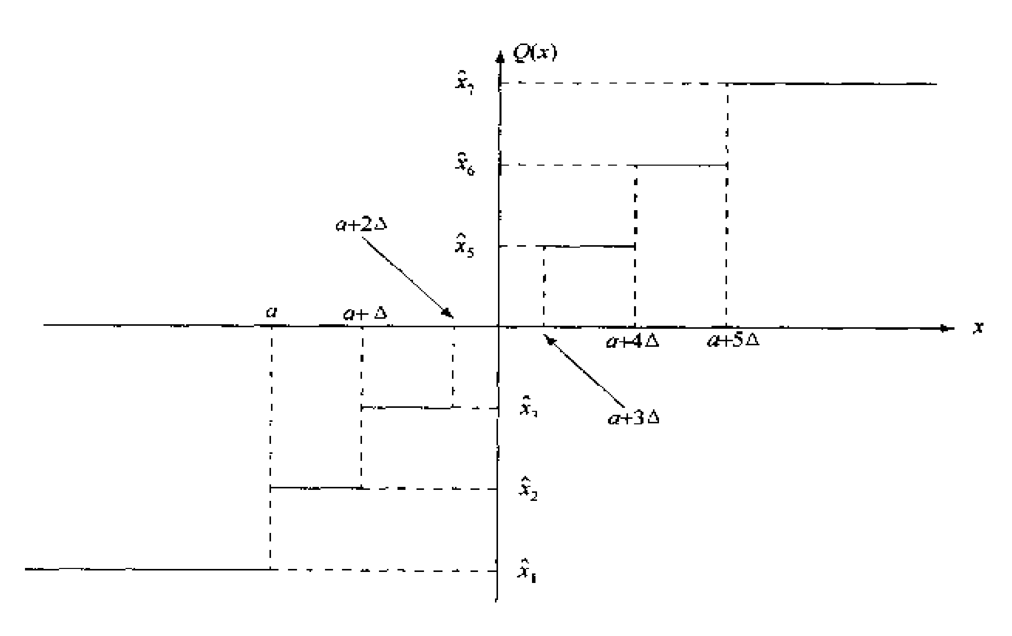


图 4.4　$N=7$ 的量化函数(注意这里 $\hat{x}_4=0$)

对于对称概率密度函数，问题甚至变得更为简单。在这种情况下：

$$\mathcal{R}_i=\begin{cases}(a_{i-1}-a_i), & 1\leqslant i\leqslant N-1\\ (a_{i-1},a_N), & i=N\end{cases}\tag{4.3.5}$$

其中,

$$\begin{cases}a_0=-\infty\\ a_i=(i-N/2)\Delta, & 1\leqslant i\leqslant N-1\\ a_N=\infty\end{cases}\tag{4.3.6}$$

这时,为了实现最小失真,仅有一个参数 Δ 需要选取。centroid.m,mse_dist.m 和 uq_dist.m 这三个 m 文件分别求一个区域的质心,已知分布和已知量化区域边界时的均方量化误差以及当采用一个均匀量化器对给定信源进行量化时的失真(假设量化电平设在量化区域的质心)。为了使用这三个 m 文件中的每一个,信源的分布(依赖于至多三个参数)必须在某个 m 文件中给出。这些 m 文件如下所示。

m 文件

```
function y=centroid(funfcn,a,b,tol,p1,p2,p3)
% CENTROID     Finds the centroid of a function over a region
%              Y=CENTROID('F',A,B,TOL,P1,P2,P3) finds the centroid of the
%              function F defined in an m-file on the [A,B] region. The
%              function can contain up to three parameters, P1, P2, P3.
%              tol=the relative error.

args=[ ];
for n=1:nargin-4
  args=[args,',p',int2str(n)];
end
args=[args,')'];
funfcn1='x_fnct';
y1=eval(['quad(funfcn1,a,b,tol,[],funfcn',args]);
y2=eval(['quad(funfcn,a,b,tol,[]',args]);
y=y1/y2;
```

m 文件

```
function [y,dist]=mse_dist(funfcn,a,tol,p1,p2,p3)
%MSE_DIST      returns the mean-squared quantization error.
%              [Y,DIST]=MSE_DIST(FUNFCN,A,TOL,P1,P2,P3)
%              funfcn=The distribution function given
%              in an m-file. It can depend on up to three
%              parameters, p1,p2,p3.
%              a=the vector defining the boundaries of the
%              quantization regions. (Note: [a(1),a(length(a))]
%              is the support of funfcn.)
%              p1,p2,p3=parameters of funfcn
%              tol=the relative error.

args=[ ];
for n=1:nargin-3
  args=[args,',p',int2str(n)];
end
args=[args,')'];
for i=1:length(a)-1
  y(i)=eval(['centroid(funfcn,a(i),a(i+1),tol',args]);
end
```

```
  dist=0;
for i=1:length(a)-1
  newfun = 'x_a2_fnct' ;
  dist=dist+eval(['quad(newfun,a(i),a(i+1),tol,[],funfcn,', num2str(y(i)), args]);
end
```

m 文件

```
function [y,dist]=uq_dist(funfcn,b,c,n,delta,s,tol,p1,p2,p3)
%UQ_DIST        returns the distortion of a uniform quantizer
%               with quantization points set to the centroids
%               [Y,DIST]=UQ_DIST(FUNFCN,B,C,N,DELTA,S,TOL,P1,P2,P3)
%               funfcn=source density function given in an m-file
%               with at most three parameters, p1,p2,p3.
%               [b,c]=The support of the source density function.
%               n=number of levels.
%               delta=level size.
%               s=the leftmost quantization region boundary.
%               p1,p2,p3=parameters of the input function.
%               y=quantization levels
%               dist=distortion.
%               tol=the relative error.

if (c-b<delta*(n-2))
  error('Too many levels for this range.'); return
end
if (s<b)
  error('The leftmost boundary too small.'); return
end
if (s+(n-2)*delta>c)
  error('The leftmost boundary too large.'); return
end
args=[];
for j=1:nargin-7
  args=[args,',p',int2str(j)];
end
args=[args,')'];
a(1)=b;
for i=2:n
  a(i)=s+(i-2)*delta;
end
a(n+1)=c;
[y,dist]=eval(['mse_dist(funfcn,a,tol',args]);
```

解说题

解说题 4.4　[质心的确定]

对一个零均值、单位方差高斯分布的量化区域求质量，量化区域的边界给出为：(-5，-4，-2，0，1，3，5)。

题　解

高斯分布由 m 文件 normal.m 给出。这个分布是两个参数的函数。这两个参数是均值和方差，分别用 m 和 s（或 σ）表示。高斯分布的支持区间是（$-\infty$，∞），但是当使用数值计算时，采用该分布标准方差的许多倍的范围也就足够了，例如（$m-10\sqrt{s}$，$m+10\sqrt{s}$）就可以使用。下面的 m 文件用于求质心（最佳量化电平）。

m 文件

```
% MATLAB script for Illustrative Problem 4.4.

echo on ;
a=[-10,-5,-4,-2,0,1,3,5,10];
for i=1:length(a)-1
  y(i)=centroid('normal',a(i),a(i+1),0.001,0,1);
  echo off ;
end
```

这个结果为如下量化电平：

(−5.1865, −4.2168, −2.3706, 0.7228, −0.4599, 1.5101, 3.2827, 5.1865)

解说题

解说题 4.5 [均方误差]

在解说题 4.4 中求均方误差。

题 解

令 $a=(-10,-5,-4,-2,0,1,3,5,10)$，用 mse_dist.m 求得均方误差为 0.177。

解说题

解说题 4.6 [均匀量化器失真]

一个方差为 4 的零均值高斯源，用具有 12 个量化电平，每个长度为 1 的均匀量化器量化，求均方误差。假设量化区域对于分布的方差是对称的。

题 解

根据对称的假设，量化区域的边界是 0, ±1, ±2, ±3, ±4 和 ±5，而量化区域是 $(-\infty,-5]$ $(-5,-4]$, $(-4,-3]$, $(-3,-2]$, $(-2,-1]$, $(-1,-0]$, $(0,1]$, $(1,2]$, $(2,3]$, $(3,4]$, $(4,5]$ 和 $(5,+\infty]$。这意味着在 uq_dis.m 函数中，能用 $b=-20, c=20, \Delta=1, n=12, s=-5, tol=0.001, p_1=0$ 和 $p_2=2$ 代入，代入这些值后求出平方误差失真为 0.0851，各量化值为 ±0.4897, ±1.4691, ±2.4487, ±3.4286, ±4.4089 和 ±5.6455。

m 文件 uq _ mdpnt.m 用于当量化电平选为量化间隔的中点时，对一个对称密度函数求平方误差失真。这时，对应于第一个和最后一个量化区域的量化电平就选为距离最外两个量化边界的 $\Delta/2$ 点处。这就是说，如果量化电平数是偶数，那么量化边界就是 $0, \pm\Delta, \pm2\Delta, \cdots, \pm(N/2-1)\Delta$，量化电平就由 $\pm\Delta/2, \pm3\Delta/2, \cdots, (N-1)\Delta/2$ 给出。如果量化电平数是奇数，那么量化边界就是 $\pm\Delta/2, \pm3\Delta/2, \cdots, \pm(N/2-1)\Delta$，量化电平由 $0, \pm\Delta, \pm2\Delta, \cdots, (N-1)\Delta/2$ 给出，这个 m 文件 uq _ mdpnt.m 如下所示。

m 文件

```
function dist=uq_mdpnt(funfcn,b,n,delta,tol,p1,p2,p3)
%UQ_MDPNT    returns the distortion of a uniform quantizer
%            with quantization points set to the midpoints
%            DIST=UQ_MDPNT(FUNFCN,B,N,DELTA,TOL,P1,P2,P3).
%            funfcn=source density function given in an m-file
%            with at most three parameters, p1,p2,p3. The density
```

```
%              function is assumed to be an even function.
%              [-b,b]=the support of the source density function.
%              n=number of levels.
%              delta=level size.
%              p1,p2,p3=parameters of the input function.
%              dist=distortion.
%              tol=the relative error.

if (2*b<delta*(n-1))
  error('Too many levels for this range.'); return
end
args=[];
for j=1:nargin-5
  args=[args,',p',int2str(j)];
end
args=[args,')'];
a(1)=-b;
a(n+1)=b;
a(2)=-(n/2-1)*delta;
y(1)=a(2)-delta/2;
for i=3:n
  a(i)=a(i-1)+delta;
  y(i-1)=a(i)-delta/2;
end
y(n)=a(n)+delta;
dist=0;
for i=1:n
  newfun = 'x_a2_fnct' ;
  dist=dist+eval(['quad(newfun,a(i),a(i+1),tol,[],funfcn,', num2str(y(i)), args]);
end
```

解说题

解说题 4.7　[量化电平设置在中点的均匀量化器]

当用一个均匀量化器对一个零均值、单位方差的高斯随机变量量化时，求失真。量化电平数是 11，每个量化区域的长度是 1。

题　解

在 uq_mdpnt.m 中，代入该密度函数的名'normal'[①]。密度函数的参数 $p_1=0, p_2=1$，量化电平数 $n=11$，量化电平长度 $\Delta=1$，参数 b（选择该密度函数的支撑区间）用 $b=10p_2=10$，并选容差为 0.001，得到的失真是 0.0833。

非均匀量化

在非均匀量化中，除了第一个和最后一个量化区域以外，每个量化区域要求等长放宽，而且可以具有任意长度。因为在这种情况下量化是在更为宽松的条件下完成的，所以结果会明显优于均匀量化。这时称为 **Lloyd-Max 条件**的最优化条件可以表示为：

① 函数的名应该用'normal'代入，包括单引号。

$$\begin{cases} \hat{x}_i = \dfrac{\int_{a_{i-1}}^{a_i} x f_X(x)\mathrm{d}x}{\int_{a_{i-1}}^{a_i} f_X(x)\mathrm{d}x} \\ a_i = \dfrac{(\hat{x}_{i-1} + \hat{x}_i)}{2} \end{cases} \tag{4.3.7}$$

从这些式子中可以得出,最优量化电平是量化区域的质心,量化区域之间的最佳边界是量化电平之间的中间值。为了求得这个 Lloyd-Max 方程的解,我们由一组量化电平 $\hat{x}_i$ 开始。由这一组值就能简单求得一组量化区域的边界 a_i,再由这组 a_i 又可以得到一组新的量化电平。这个过程一直继续到从一次迭代到另一次迭代时失真再也没有明显改善为止。这个算法保证收敛到某个局部最小值,但一般不保证可实现全局最小值。

下面的 m 文件 lloydmax.m 给出了设计一个最优量化器的程序。

m 文件

```
function [a,y,dist]=lloydmax(funfcn,b,n,tol,p1,p2,p3)
%LLOYDMAX     returns the the Lloyd-Max quantizer and the mean-squared
%             quantization error for a symmetric distribution
%             [A,Y,DIST]=LLOYDMAX(FUNFCN,B,N,TOL,P1,P2,P3).
%             funfcn=the density function given
%             in an m-file. It can depend on up to three
%             parameters, p1,p2,p3.
%             a=the vector giving the boundaries of the
%             quantization regions.
%             [-b,b] approximates support of the density function.
%             n=the number of quantization regions.
%             y=the quantization levels.
%             p1,p2,p3=parameters of funfcn.
%             tol=the relative error.

args=[];
for j=1:nargin-4
  args=[args,',p',int2str(j)];
end
args=[args,')'];
v=eval(['variance(funfcn,-b,b,tol',args]);
a(1)=-b;
d=2*b/n;
for i=2:n
  a(i)=a(i-1)+d;
end
a(n+1)=b;
dist=v;
[y,newdist]=eval(['mse_dist(funfcn,a,tol',args]);
while(newdist<0.99*dist),
  for i=2:n
    a(i)=(y(i-1)+y(i))/2;
  end
  dist=newdist;
  [y,newdist]=eval(['mse_dist(funfcn,a,tol',args]);
end
```

解说题

解说题 4.8　[Lloyd-Max 量化器设计]

为一个零均值、单位方差的高斯源设计一个 10 电平的 Lloyd-Max 量化器。

题　解

在 lloydmax.m 中采用 $b=10, n=10, tol=0.01, p_1=0$ 和 $p_2=1$，得出量化边界和量化电平向量 $\boldsymbol{a}$ 和 $\boldsymbol{y}$ 分别为：

$$\boldsymbol{a}=(\pm 10, \pm 2.16, \pm 1.51, \pm 0.98, \pm 0.48, 0)$$

$$\boldsymbol{y}=(\pm 2.52, \pm 1.78, \pm 1.22, \pm 0.72, \pm 0.24)$$

所得失真是 0.02。这些都是 Max[2]的表中给出的最优值的好的近似。

4.3.2　矢量量化

在标量量化中，离散时间信息源的每个输出被量化，然后进行编码。例如，如果用一个 4 电平的标量量化器，并将每个电平编码到 2 比特，那么每个信源输出就要用 2 个比特。这种量化方法如图 4.5 所示。

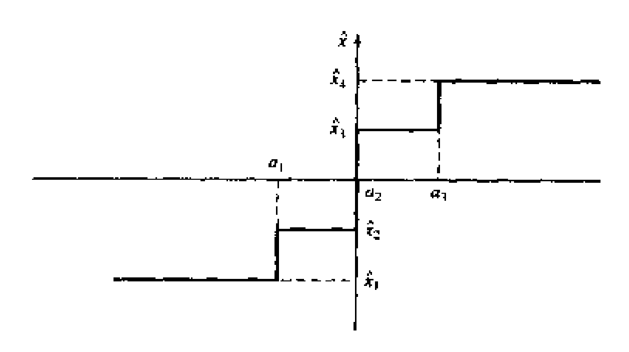


图 4.5　4 电平标量量化器

现在，如果每次考虑两个信息源样本，并将这两个样本看成是在某个平面上的一个点，那么这个标量量化器就将整个平面划分成 16 个量化区域，如图 4.6 所示。

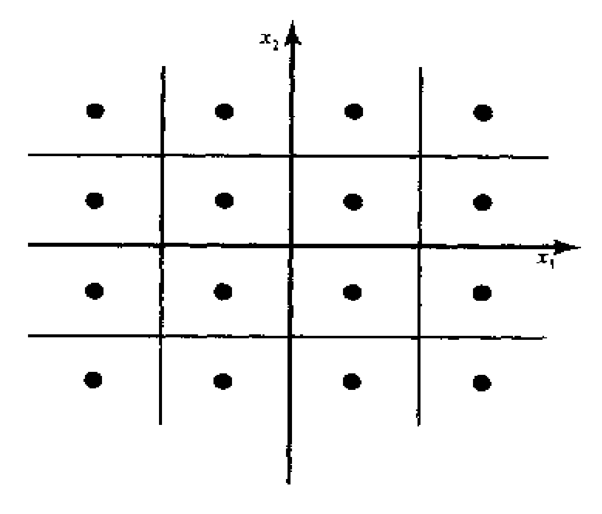


图 4.6　用于两个样本的标量 4 电平量化

可以看到,在二维空间内这些量化区域全都是矩形形状的。如果在二维空间上允许任意形状的 16 个区域,还能够获得更好的结果。这就意味着,用 16 个量化区域一次对两个信源输出进行量化,等效于每两个信源输出为 4 比特或每个信源输出的比特数等于在标量量化中所获得的每个信源输出的比特数。由于放宽了具有矩形量化区域的要求,性能还可以进一步得到改善。现在,如果一次取 3 个样本,并将整个三维空间量化为 64 个量化区域,那么甚至比用每个信源输出相同的比特数还能有更小的失真。矢量量化的概念是取长度为 n 的信息源组,并在 n 维的欧氏空间内设计量化器,而不是在一维空间内根据单个样本进行量化。

假设在 n 维空间的量化区域记为$\mathcal{R}_i$, $1\leqslant i\leqslant K$,这 K 个量化区域将 n 维空间进行剖分。长为 n 的每个信源输出组记为 $\mathbf{x}\in\mathbb{R}^n$,并且若 $\mathbf{x}\in\mathcal{R}_i$,它就被量化到$\mathcal{Q}(\mathbf{x})=\hat{\mathbf{x}}_i$。图 4.7 显示的是 $n=2$ 时的这种量化方法。现在,因为总共有 K 个量化值,用 $\log_2 K$ 比特就足以表示这些值,这就是说,每 n 个信源输出需要 $\log_2 K$ 个比特,或者信源码率是:

$$R=\frac{\log_2 K}{n}\qquad \text{比特数/信源输出} \tag{4.3.8}$$

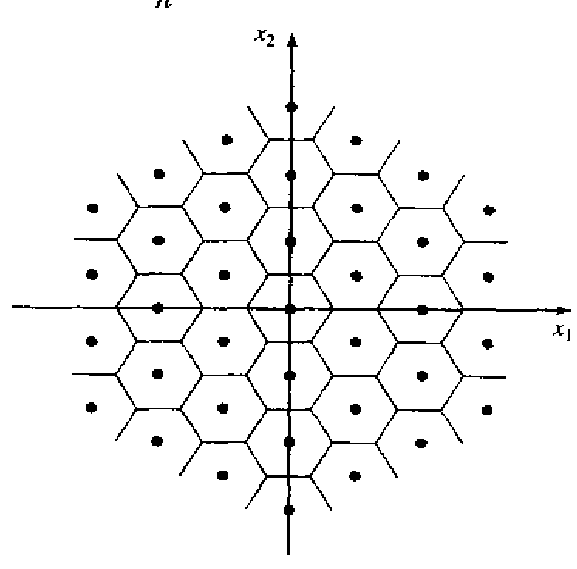


图 4.7 二维矢量量化

n 维并且电平数为 K 的最优矢量量化器是这样一种量化器,它选取量化区域$\mathcal{R}_i$ 和量化值 $\hat{\mathbf{x}}_i$,以使得所产生的失真最小。利用在标量量化中所采用的相同步骤,对最优矢量量化器设计可得出以下准则:

1. 在 n 维空间内,区域$\mathcal{R}_i$ 是全部这样的点的集合:对于全部 $j\neq i$,这些点比任何其他的 $\hat{\mathbf{x}}_j$ 都更接近于 $\hat{\mathbf{x}}_i$,即

$$\mathcal{R}_i=\{\mathbf{x}\in\mathbb{R}^n:\|\mathbf{x}-\hat{\mathbf{x}}_i\|<\|\hat{\mathbf{x}}-\hat{\mathbf{x}}_j\|,\ \forall j\neq i\}$$

2. $\hat{\mathbf{x}}_i$ 是域$\mathcal{R}_i$ 的质心:

$$\hat{\mathbf{x}}_i=\frac{1}{p(\mathbf{X}\in\mathcal{R}_i)}\iint\cdots\int_{\mathcal{R}_i}\mathbf{x}f_{\mathbf{X}}(\mathbf{x})\mathrm{d}\mathbf{x}$$

设计最优矢量量化器的可行途径基于设计最优标量量化器时所采用的同样的途径。从某一给定的量化区域集合入手,利用上述的准则 2 导出对于这些量化区域的最优量化矢量,然后

利用准则1重新分割量化空间，如此反复多次，直到在失真上的变化可以忽略为止。当不是用概率密度函数，而是有某个大的训练序列时，设计矢量量化器的类似方法称为广义 Lloyd 算法，K 法算法或 LBG(Linde-Buzo-Gray)算法。

令该训练集合为$\{\mathbf{x}_i\}_{i=1}^{N}$，式中 $\mathbf{x}_i \in \mathbb{R}^n$。LBG 算法用于设计一个具有 K 量化矢量和码率 $R = \log_2 K/n$ 的 n 维矢量量化器，该算法可归纳如下：

1. 选取 K 个任意量化矢量$\{\hat{\mathbf{x}}_k\}_{k=1}^{K}$作为 n 维矢量。
2. 将训练集合中的矢量剖分为子集$\{\mathcal{R}_k\}_{k=1}^{K}$，其中每个子集是那些最接近于 $\hat{\mathbf{x}}_k$ 的训练矢量的集，即

$$\mathcal{R}_k = \{\mathbf{x}_i : \|\mathbf{x}_i - \hat{\mathbf{x}}_k\| < \|\mathbf{x}_i - \hat{\mathbf{x}}_{k'}\|, \qquad k' \neq k\}$$

3. 通过$\mathcal{R}_k$ 的质心更新量化矢量，将在$\mathcal{R}_k$ 中的训练矢量的个数记为 $\|\mathcal{R}_k\|$，求得更新后的量化矢量为：

$$\hat{\mathbf{x}}_k = \frac{1}{\|\mathcal{R}_k\|} \sum_{\mathbf{x} \in \mathcal{R}_k} \mathbf{x}$$

4. 计算失真。如果在最后一步失真没有显著变化就终止，否则再回到第 2 步。

采用 LBG 算法计算矢量量化器的 m 文件 vq.m 如下所示。

m 文件

```
function [codebook,distortion]=VQ(training_seq,dimension,codebook_size,tolerance)
%VQ.m   vector quantizer design using K-means algorithm
%       [codebook,distortion]=VQ(training_seq,dimension,codebook_size,tolerance).
%       training_seq = training sequence.
%       dimension = dimension of the quantizer.
%       codebook_size = size of the codebook (rate=log2(codebook_size)/dimension).
%       tolerance = desired relative distortion (default=0.001).
%       Length of training_seq must be a multiple of the dimension of quantizer.

if (nargin==3)
  tolerance=0.001;
end
m=round(length(training_seq)/dimension);
if (m*dimension-length(training_seq)<0)
  error('length of training_seq is not a multiple of dimension')
end
%       Initialize the codebook.
initial=training_seq(1:dimension*codebook_size);
initialcodebook=(reshape(initial,dimension,codebook_size))';
updated_codebook=initialcodebook;
%       first update
newdistortion=0;
distortion=0;
for i=1:m;
  training_seq_block=training_seq((i-1)*dimension+1:i*dimension);
  training_matrix(i,:)=training_seq_block;
  distortion_block=[ ];
  for j=1:codebook_size;
    distort=sum((training_seq_block-updated_codebook(j,:)).^2);
    distortion_block=[distortion_block distort];
  end
  [distortion_min,ind]=min(distortion_block);
  newdistortion=newdistortion+distortion_min;
  index(i)=ind;
```

```
end
for l=1:codebook_size;
  partition=(index==l);
  if sum(partition)>0
    updated_codebook(l,:)=partition*training_matrix/sum(partition);
  end
end
newdistortion=newdistortion/m;
%      furthur updated until the desired tolerance is met
while(abs(distortion-newdistortion)/newdistortion>tolerance)
  distortion=newdistortion;
  newdistortion=0;
  for i=1:m;
    training_seq_block=training_seq((i-1)*dimension+1:i*dimension);
    training_matrix(i,:)=training_seq_block;
    distortion_block=[ ];
    for j=1:codebook_size;
      distort=sum((training_seq_block-updated_codebook(j,:)).^2);
      distortion_block=[distortion_block distort];
    end
    [distortion_min,ind]=min(distortion_block);
    newdistortion=newdistortion+distortion_min;
    index(i)=ind;
  end
  for l=1:codebook_size,
    partition=(index==l);
    if sum(partition)>0
      updated_codebook(l,:)=partition*training_matrix/sum(partition);
    end
  end
  newdistortion=newdistortion/m;
end
codebook=updated_codebook;
distortion=newdistortion/dimension;
```

解说题

解说题 4.9　[一个高斯信息源的 VQ]

利用 LBG 算法对一个均值为 0,方差为 1 的高斯信息源设计码率为 0.5,1,1.5,2,2.5 和 3 的二维矢量量化器。画出产生的信源输出矢量和量化点,并确定失真。

题　解

利用下面一组命令对 $R=1$ 产生出 36 000 个高斯型输出,这对应于 18 000 个输出矢量。

```
>> r=1;
>> k=2;
>> n=2^(r*k);
>> [C1,D1]=vq(x,k,n,0.001);
>> plot(x(:,1:2:36000),x(:,2:2:36000),'.Y',C1(:,1),C1(:,2),'*K'')
```

对于其余的 r 值也能产生类似的图。图 4.8 给出了若干样本图。

解说题

解说题 4.10　[一个高斯－马尔可夫信息源的 VQ]

对一个 $\rho=0.9$ 的高斯－马尔可夫信息源,码率 $R=0.5,1,1.5,2$ 和 2.5 时重做解说题 4.9。由高斯信息源通过关系 $y_i=\rho y_{i-1}+x_{i-1}$(式中 x_i 是高斯序列)可以得到高斯－马尔可夫序列。

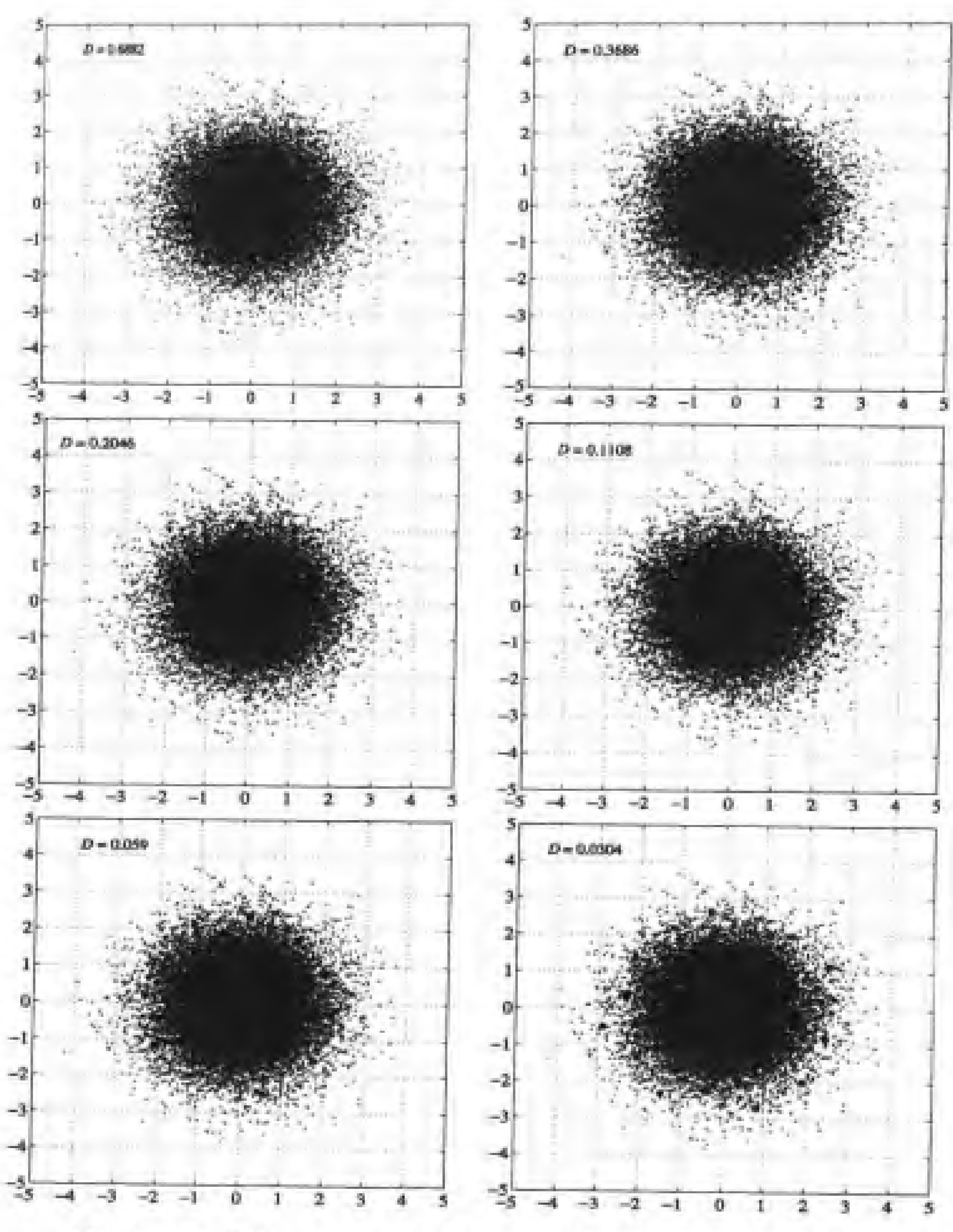


图 4.8　$k=2$，码率为 $R=0.5,1,1.5,2,2.5$ 和 3 的高斯信息源的矢量量化仿真

题　解

再次用 $R=1$，有

```
>> x = randn(1,36000);
>> y(1) = 0;
```

```
>> for i=1:36000
   y(i+1)=0.9*y(i)+x(i);
   end
>> y=y(2:36001);
>> r=1;
>> k=2;
>> n=2^(r*k);
>> [C1,D1]=vq(y,k,n,0.001);
```

图 4.9 给出了所对应的图。

在语音和图像编码中,矢量量化已经获得广泛应用,并为了降低它的计算复杂性业已提出很多算法。对于平稳和各态遍历的信源来说,可以证明,随着 n 的增大,矢量量化器的性能接近于由率失真函数给出的最佳性能。

4.3.3 脉冲编码调制

在脉冲编码调制中,模拟信号首先以高于奈奎斯特率的速率采样,然后将所得样本量化。假设模拟信号是在以$[-x_{max}, x_{max}]$表示的区间内分布的,而量化电平数很大。量化电平可以是相等的或是不相等的;前者属于均匀 PCM,而后者属于非均匀 PCM。

4.3.4 均匀 PCM

在均匀 PCM 中,长度为 $2x_{max}$ 的区间$[-x_{max}, x_{max}]$被划分为 N 个相等的子区间,每个区间的长度为 $\Delta=2x_{max}/N$。如果 N 足够大,那么在每个子区间内输入的密度函数就可以认为是均匀的,产生的失真为 $D=\Delta^2/12$。如果 N 是 2 的幂次方,即 $N=2^v$,那么就要求用 v 比特来表示每个量化电平。这就意味着,如果模拟信号的带宽是 W,采样又是在奈奎斯特速率下完成的,那么传输 PCM 信号所要求的带宽至少是 vW(实际上 $1.5vW$ 比较接近于实际情况)。这时的失真由下式给出:

$$D=\frac{\Delta^2}{12}=\frac{x_{max}^2}{3N^2}=\frac{x_{max}^2}{3\times 4^v} \tag{4.3.9}$$

如果模拟信号的功率用$\overline{X^2}$表示,则信号/量化噪声的比(SQNR)由下式给出:

$$\begin{aligned}\text{SQNR}&=3N^2\frac{\overline{X^2}}{x_{max}^2}\\&=3\times 4^v\frac{\overline{X^2}}{x_{max}^2}\\&=3\times 4^v\ \overline{\breve{X}^2}\end{aligned} \tag{4.3.10}$$

其中$\breve{X}$表示归一化输入,定义为:

$$\breve{X}=\frac{X}{x_{max}}$$

以分贝(dB)计的 SQNR 为:

$$\text{SQNR}|_{dB}\approx 4.8+6v+\overline{\breve{X}^2}|_{dB} \tag{4.3.11}$$

量化以后,这些已量化的电平用 v 比特对每个已量化电平进行**编码**。编码方法通常使用**自然二进制码**(NBC),即最低电平映射为全 0 序列,最高电平映射为全 1 序列,其余的全部电

平按已量化值的递增次序映射。

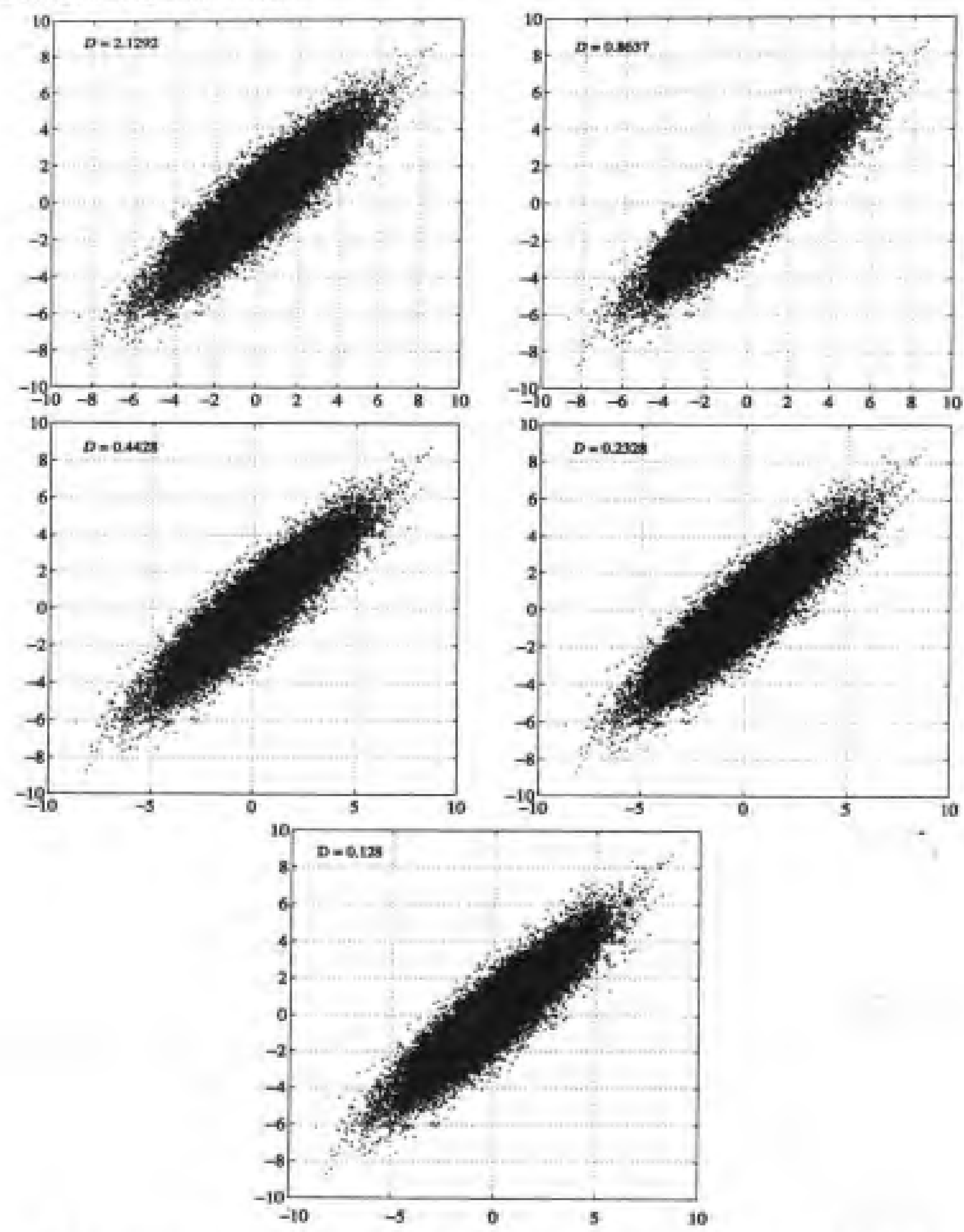


图 4.9　$k=2$，码率为 $R=0.5,1,1.5,2$ 和 2.5 的高斯－马尔可夫信息源的矢量量化仿真

下面给出的 m 文件 u_pcm.m 用采样值序列和要求的量化电平数作为输入，求得已量化序列、编码序列和产生的 SQNR（以 dB 计）。

m 文件

```
function [sqnr,a_quan,code]=u_pcm(a,n)
%U_PCM           uniform PCM encoding of a sequence
%                [SQNR,A_QUAN,CODE]=U_PCM(A,N)
%                a=input sequence.
%                n=number of quantization levels (even).
%                sqnr=output SQNR (in dB).
%                a_quan=quantized output before encoding.
%                code=the encoded output.

amax=max(abs(a));
a_quan=a/amax;
b_quan=a_quan;
d=2/n;
q=d.*[0:n-1];
q=q-((n-1)/2)*d;
for i=1:n
  a_quan(find((q(i)-d/2 <= a_quan) & (a_quan <= q(i)+d/2)))=...
  q(i).*ones(1,length(find((q(i)-d/2 <= a_quan) & (a_quan <= q(i)+d/2))));
  b_quan(find( a_quan==q(i) ))=(i-1).*ones(1,length(find( a_quan==q(i) )));
end
a_quan=a_quan*amax;
nu=ceil(log2(n));
code=zeros(length(a),nu);
for i=1:length(a)
  for j=nu:-1:0
    if ( fix(b_quan(i)/(2^j)) == 1)
        code(i,(nu-j)) = 1;
        b_quan(i) = b_quan(i) - 2^j;
    end
  end
end
sqnr=20*log10(norm(a)/norm(a-a_quan));
```

解说题

解说题 4.11　[均匀 PCM]

产生一个幅度为 1 和 $\omega=1$ 的正弦信号,用均匀 PCM 方法分别用 8 电平和 16 电平进行量化,在同一个坐标轴上画出原信号和已量化信号,比较这两种情况下的 SQNR。

题　解

任意选取信号持续期为 10 s,然后用 m 文件 u_pcm.m 产生 8 量化电平和 16 量化电平两种情况下的已量化信号,所得的 SQNR 是:8 电平 PCM 为 18.90 dB,16 电平均匀 PCM 为 25.13 dB,图 4.10 给出了对应的图。

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 4.11.
echo on
t=[0:0.01:10];
a=sin(t);
[sqnr8,aquan8,code8]=u_pcm(a,8);
[sqnr16,aquan16,code16]=u_pcm(a,16);
pause    % Press a key to see the SQNR for N = 8.
```

```
sqnr8
pause    % Press a key to see the SQNR for N = 16
sqnr16
pause    % Press a key to see the plot of the signal and its quantized versions.
plot(t,a,'-',t,aquan8,'-.',t,aquan16,'-',t,zeros(1,length(t)))
```

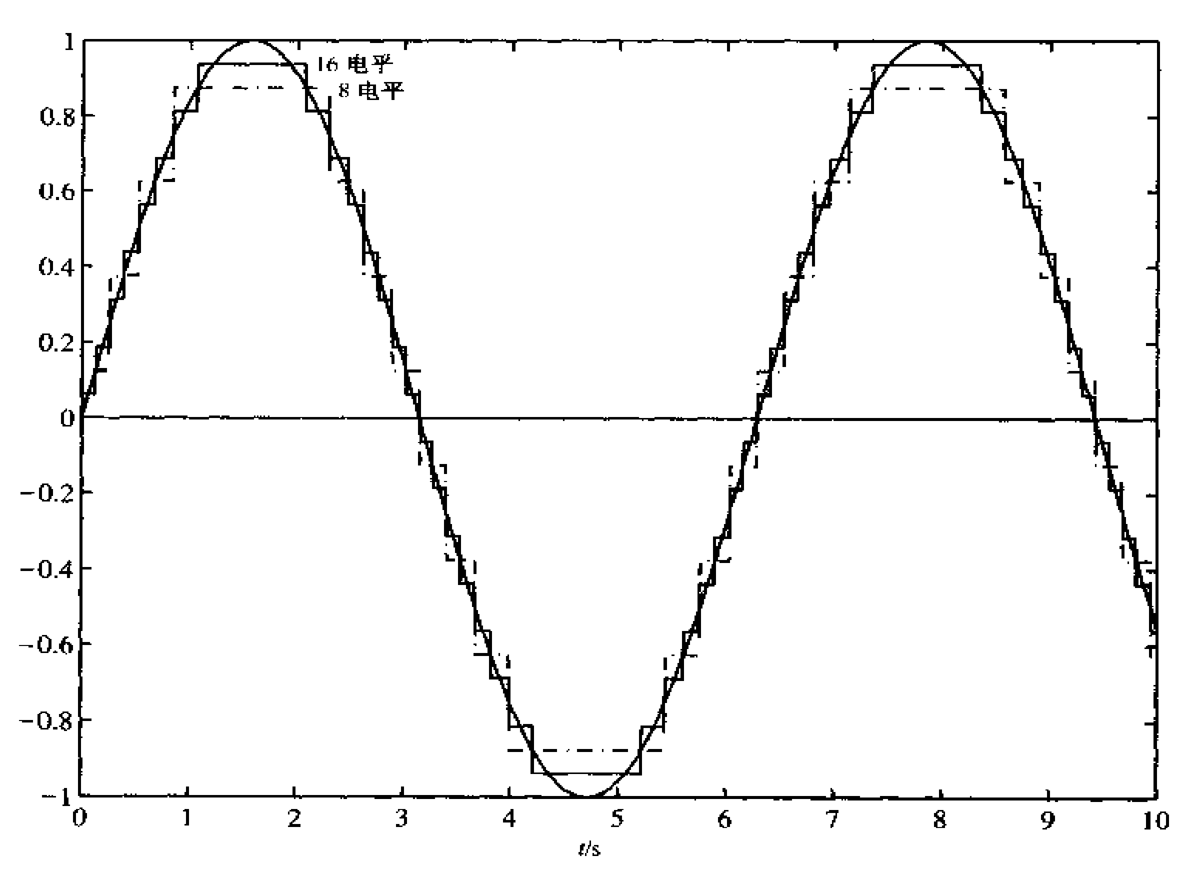


图 4.10　用 8 电平和 16 电平对一个正弦信号进行的均匀 PCM

解说题

解说题 4.12　[均匀 PCM]

产生长度为 500 的零均值、单位方差的高斯随机变量序列，利用 u_pcm.m 求当量化电平数为 64 时所得的 SQNR。求出该序列的前 5 个值、相应的量化值和相应的码字。

题　解

下面的 m 文件给出了本题的解。

m 文件

```
% MATLAB script for Illustrative Problem 4.12.

echo on
a=randn(1,500);
n=64;
```

```
[sqnr,a_quan,code]=u_pcm(a,64);
pause    % Press a key to see the SQNR.
sqnr
pause    % Press a key to see the first five input values.
a(1:5)
pause    % Press a key to see the first five quantized values.
a_quan(1:5)
pause    % Press a key to see the first five codewords.
code(1:5,:)
```

在这个 m 文件的一般运行中，可以观察到下面这些值：

$$\text{SQNR} = 31.66\ \text{dB}$$

$$输入 = (0.1775, -0.4540, 1.0683, -2.2541, 0.5376)$$

$$量化值 = (0.1569, -0.4708, 1.0985, -2.2494, 0.5754)$$

$$码字 = \begin{cases} 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 \end{cases}$$

要注意，这个程序在各次运行中会产生不同的输入值、量化值和码字，但所得的 SQNR 是十分接近的。

解说题

解说题 4.13 [量化误差]

在解说题 4.12 中，画出量化误差(定义为输入值和量化值之间的差)，同时也画出量化值作为输入值的函数的图。

题 解

图 4.11 给出了所要求的两个图。

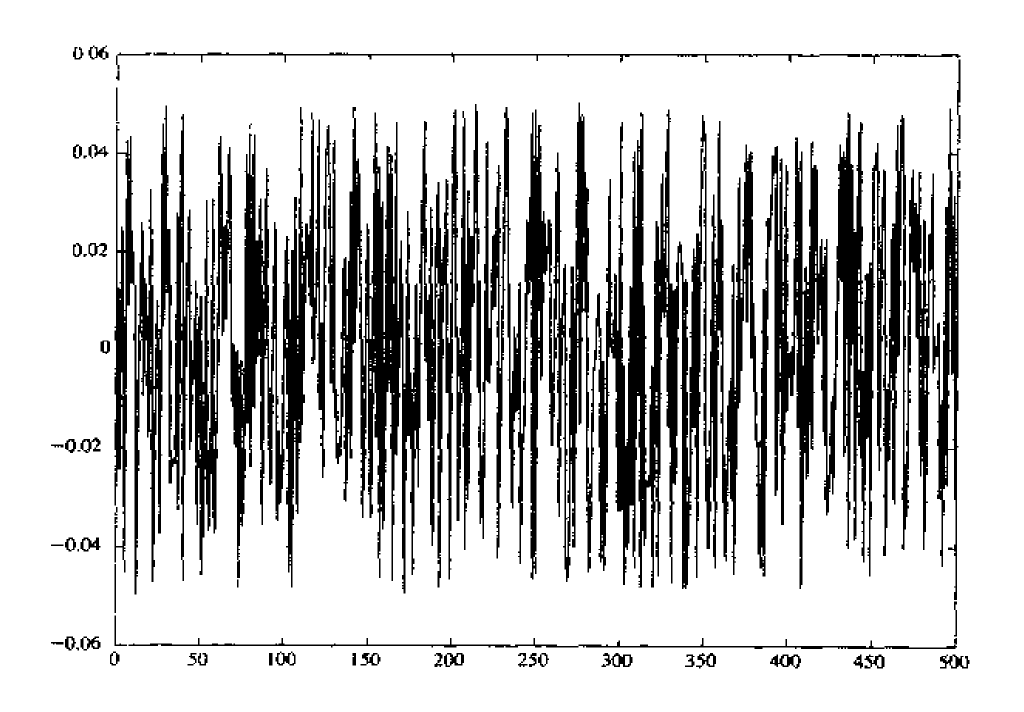
解说题

解说题 4.14 [量化误差]

分别用 16 和 128 的量化电平数重做解说题 4.13，并比较结果。

题 解

16 量化电平的结果如图 4.12 所示，128 量化电平的结果如图 4.13 所示。由图 4.11，图 4.12和图 4.13 明显可以看出，量化电平数愈大，量化误差愈小，这与我们所期望的是一致的。另外还要注意，对于大的量化电平数，输入和量化值之间的关系趋近于通过原点的斜率为 1 的一条直线，这表明输入和量化值几乎是相等的。而对于小的量化电平数(如 16)，这个关系离相等甚远，如图 4.12 所示。



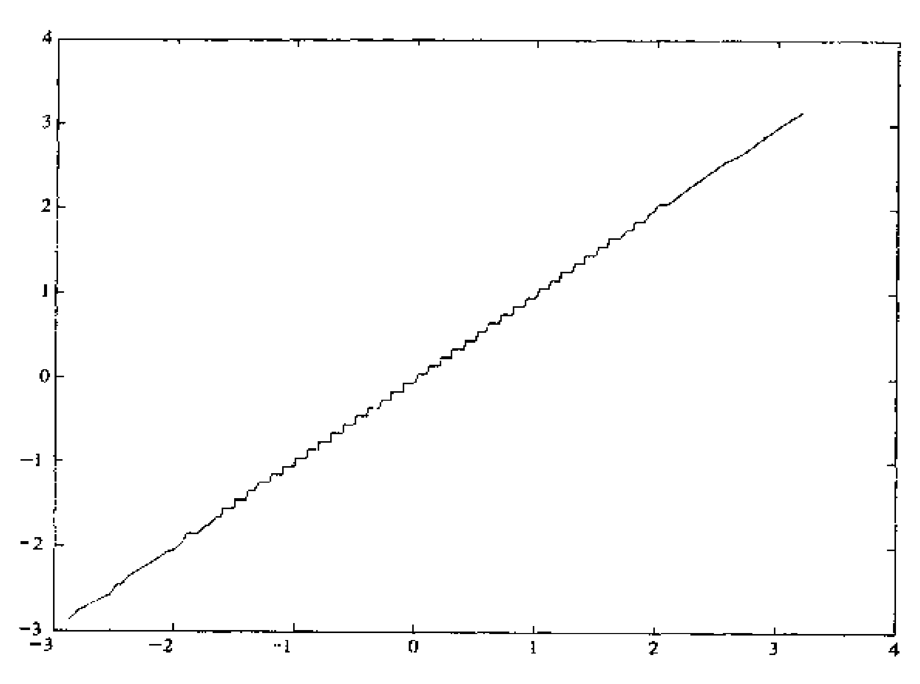
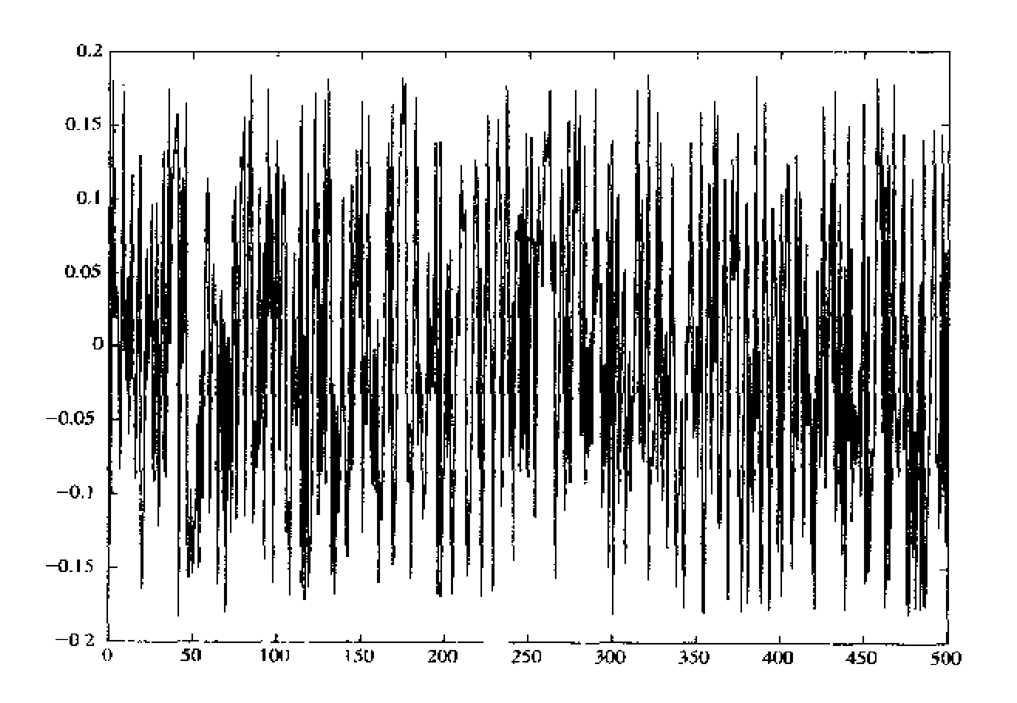


图 4.11　64 量化电平时均匀 PCM 的量化误差



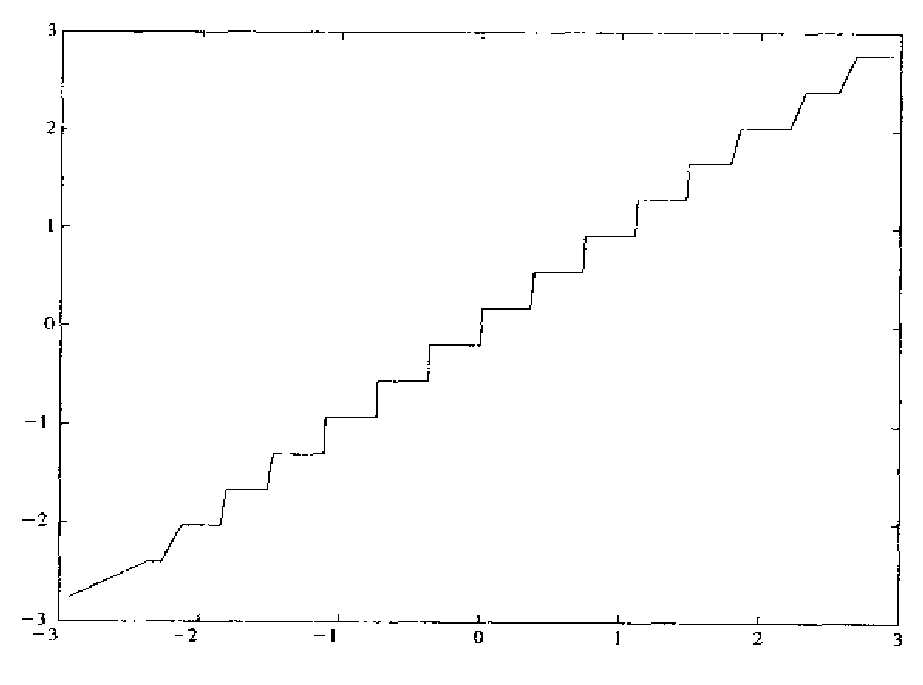
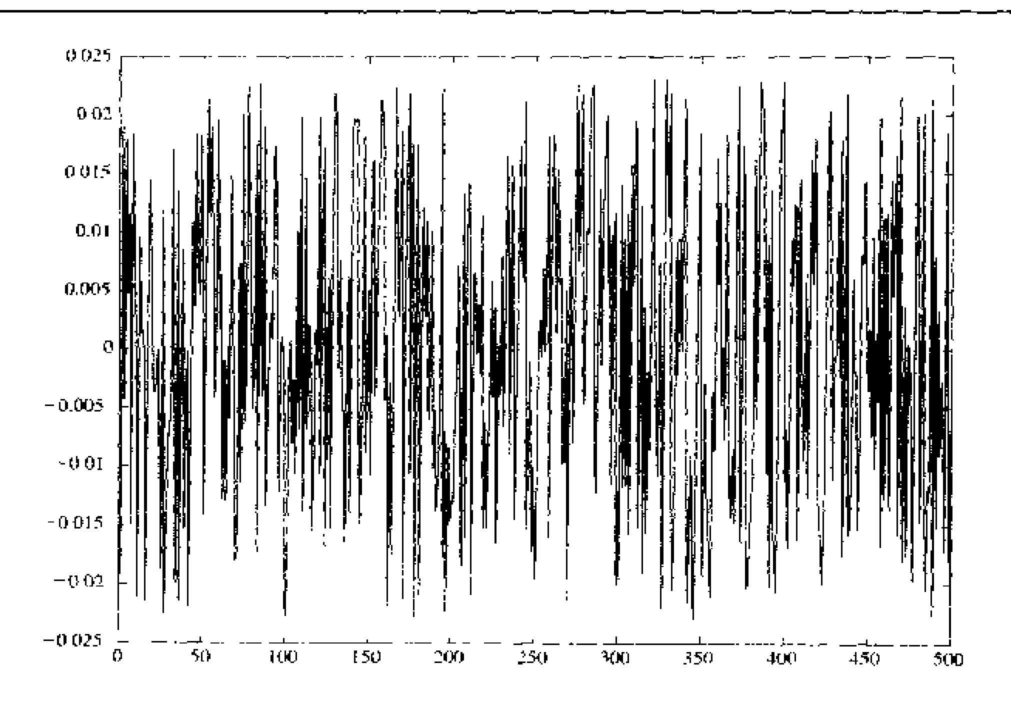


图 4.12　16 量化电平数的量化误差



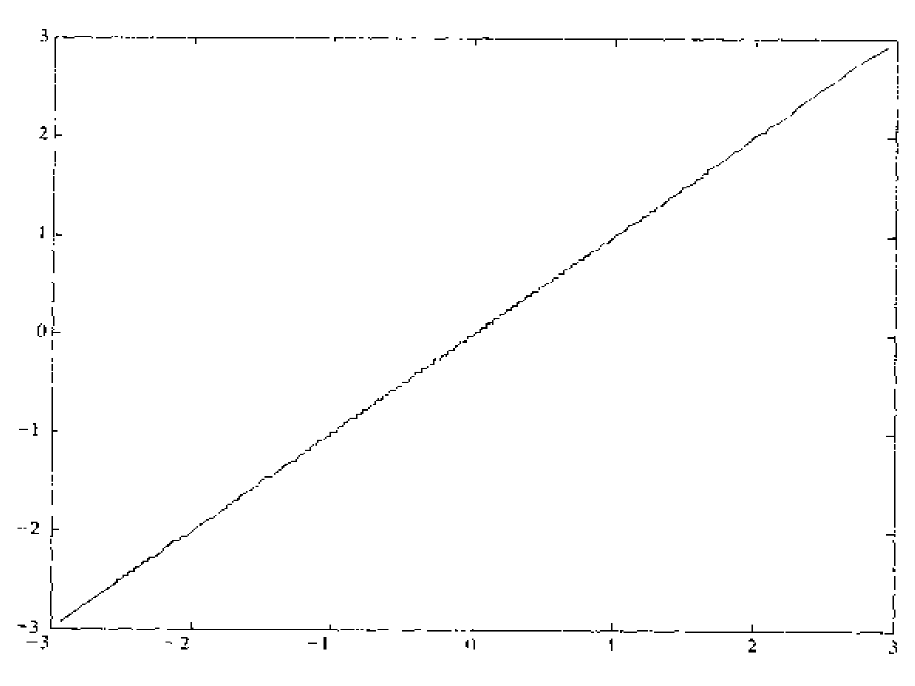


图 4.13　128 量化电平数的量化误差

非均匀 PCM

在非均匀 PCM 中,输入信号首先通过一个非线性环节,以减小输入的动态范围,再将输出加到某个均匀 PCM 系统上。在接收端,输出再通过另一个非线性环节,该环节是在发送端所用的非线性环节的逆特性。这样,总的效果就等效于一个在量化电平之间具有非均匀间隔的 PCM 系统。一般情况下,对语音信号传输来说,所使用的非线性可以是 μ 律的非线性或 A 律的非线性。

一个 μ 律的非线性定义为:

$$y=g(x)=\frac{\ln(1+\mu|x|)}{\ln(1+\mu)}\operatorname{sgn}(x) \tag{4.3.12}$$

其中 x 是归一化输入($|x|\leqslant 1$),μ 是一个参数,在标准 μ 律的非线性中它等于 255。图 4.14 给出了不同 μ 律的非线性的图。

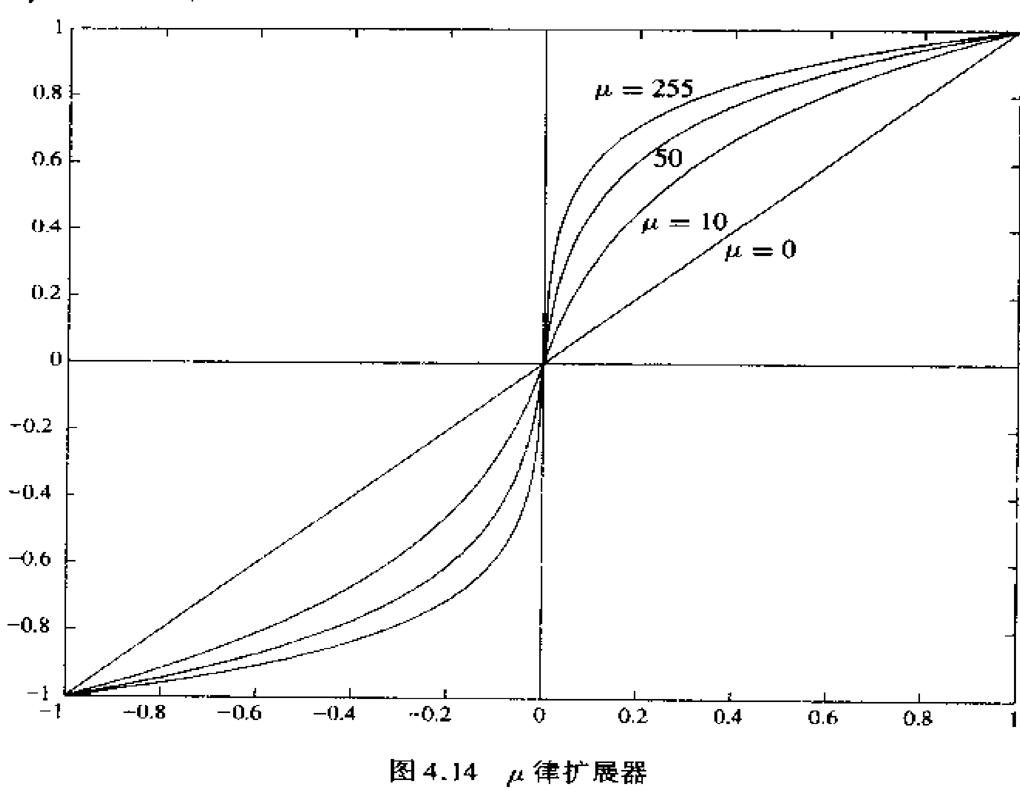


图 4.14 μ 律扩展器

μ 律非线性的逆为:

$$x=\frac{(1+\mu)^{|y|}-1}{\mu}\operatorname{sgn}(y) \tag{4.3.13}$$

以下给出的两个 m 文件 mulaw.m 和 invmulaw.m 用于实现 μ 律非线性和它的逆。

m 文件

```
function [y,a]=mulaw(x,mu)
%MULAW          mu-law nonlinearity for nonuniform PCM
%               Y=MULAW(X,MU).
%               X=input vector.

a=max(abs(x));
y=(log(1+mu*abs(x/a))./log(1+mu)).*signum(x);
```

m 文件

```
function x=invmulaw(y,mu)
%INVMULAW            the inverse of mu-law nonlinearity
%X=INVMULAW(Y,MU)    Y=normalized output of the mu-law nonlinearity.

x=(((1+mu).^(abs(y))-1)./mu).*signum(y);
```

当使用一个 μ 律 PCM 方法时，m 文件 mula _ pcm.m 就等效于 m 文件 u _ pcm.m。这个文件如下所示，用于 μ 律 PCM 系统。

m 文件

```
function [sqnr,a_quan,code]=mula_pcm(a,n,mu)
%MULA_PCM      mu-law PCM encoding of a sequence
%              [SQNR,A_QUAN,CODE]=MULA_PCM(A,N,MU).
%              a=input sequence.
%              n=number of quantization levels (even).
%              sqnr=output SQNR (in dB).
%              a_quan=quantized output before encoding.
%              code=the encoded output.

[y,maximum]=mulaw(a,mu);
[sqnr,y_q,code]=u_pcm(y,n);
a_quan=invmulaw(y_q,mu);
a_quan=maximum*a_quan;
sqnr=20*log10(norm(a)/norm(a-a_quan));
```

解说题

解说题 4.15　[非均匀 PCM]

产生一个长度为 500，按 $\mathcal{N}(0,1)$ 分布的随机变量序列。用 16，64 和 128 量化电平数和 $\mu=255$ 的 μ 律非线性，画出每种情况下量化器的误差和输入 - 输出关系，并求 SQNR。

题　解

令向量 $\boldsymbol{a}$ 是按 $\mathcal{N}(0,1)$ 产生的长度为 500 的向量，即

$$\boldsymbol{a} = \text{randn}(1,500)$$

然后，利用

$$[\text{dist}, \text{a_quan}, \text{code}] = \text{mula_pcm}(\boldsymbol{a}, 16, 255)$$

即可得到 16 量化电平数的已量化序列和 SQNR，这时的 SQNR 是 13.76 dB。对于 64 量化电平数的情况，SQNR = 25.89 dB；而对于 128 量化电平数的情况，SQNR = 31.76 dB。可见在所有情况下，将这些结果与均匀 PCM 相比，其性能都比均匀 PCM 的结果差。图 4.15，图 4.16 和图 4.17分别给出了量化器的输入 - 输出关系和量化误差。

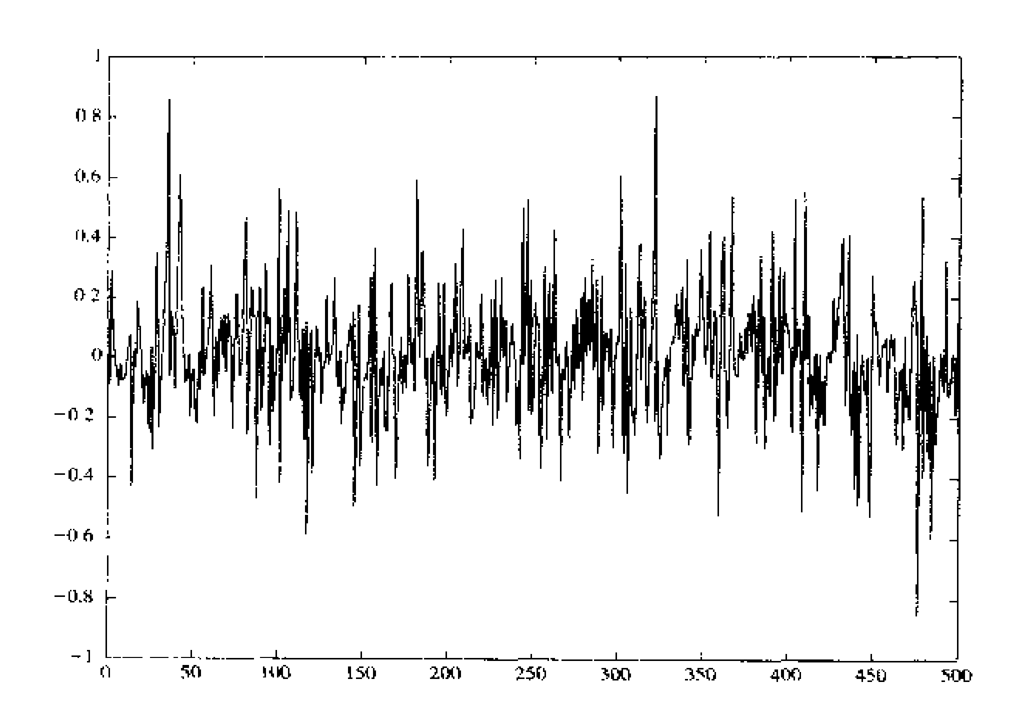
将图 4.12 和图 4.15 所示的均匀和非均匀 PCM 的输入 - 输出关系进行比较，就清楚地说明了为什么前者称为均匀 PCM，后者称为非均匀 PCM。

根据前面这个例子可以看到，在这个情况下非均匀 PCM 的性能不如均匀 PCM 的好。其原因是在这个例子中输入信号的动态范围不太大。下面这个例子说明了非均匀 PCM 的性能优于均匀 PCM 的性能。

解说题

解说题 4.16　[非均匀 PCM]

长度为 500 的非平稳序列 $\boldsymbol{a}$ 由两部分组成：前 20 个样本是按照均值为零和方差为 400 ($\sigma=20$)的高斯随机变量产生的；其余 480 个样本是根据均值为零和方差为 1 的高斯随机变量产生的。这个序列分别用均匀 PCM 和非均匀 PCM 方法量化，试比较两种情况下所得到的 SQNR。



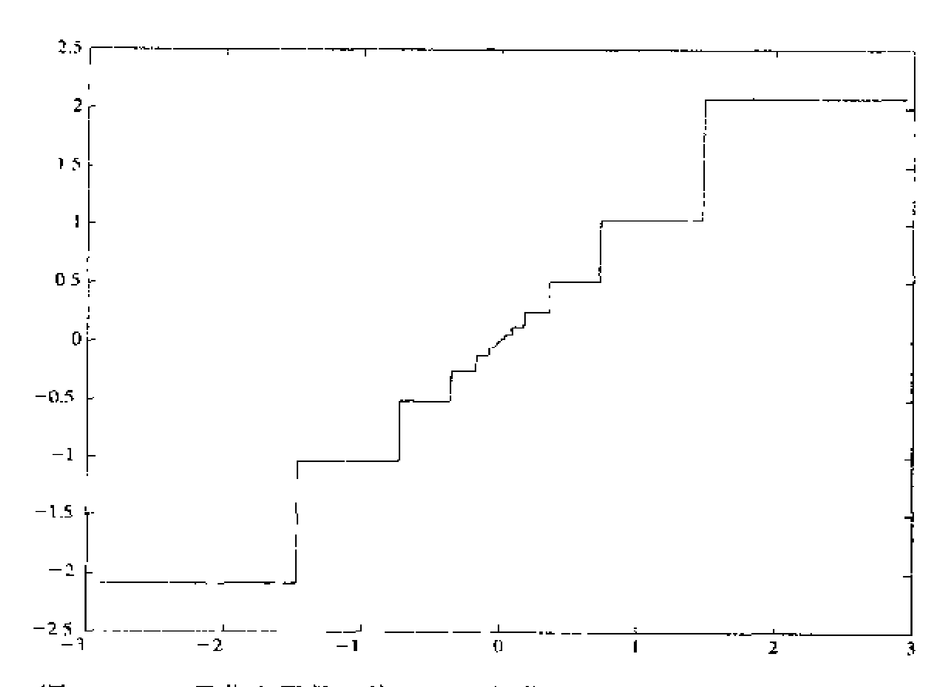
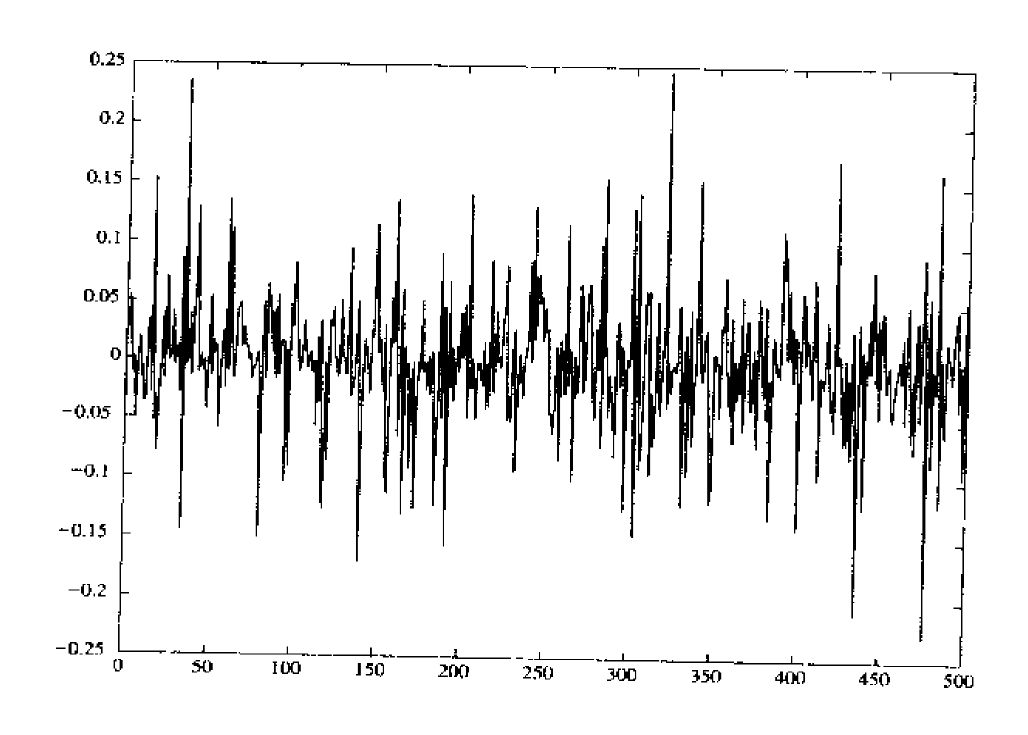


图 4.15　16 量化电平数 μ 律 PCM 的量化误差和量化器输入－输出关系



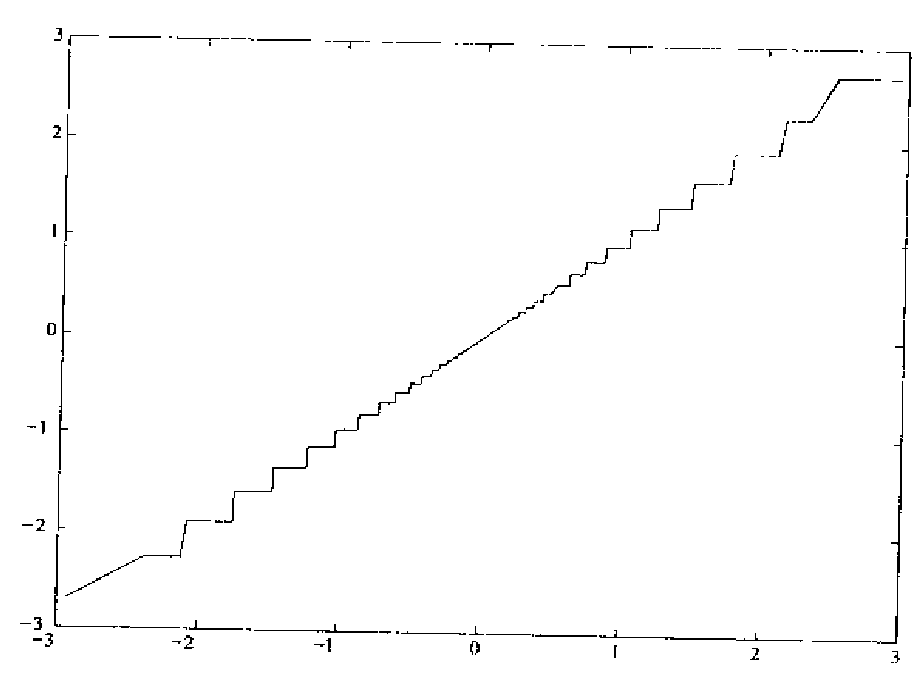
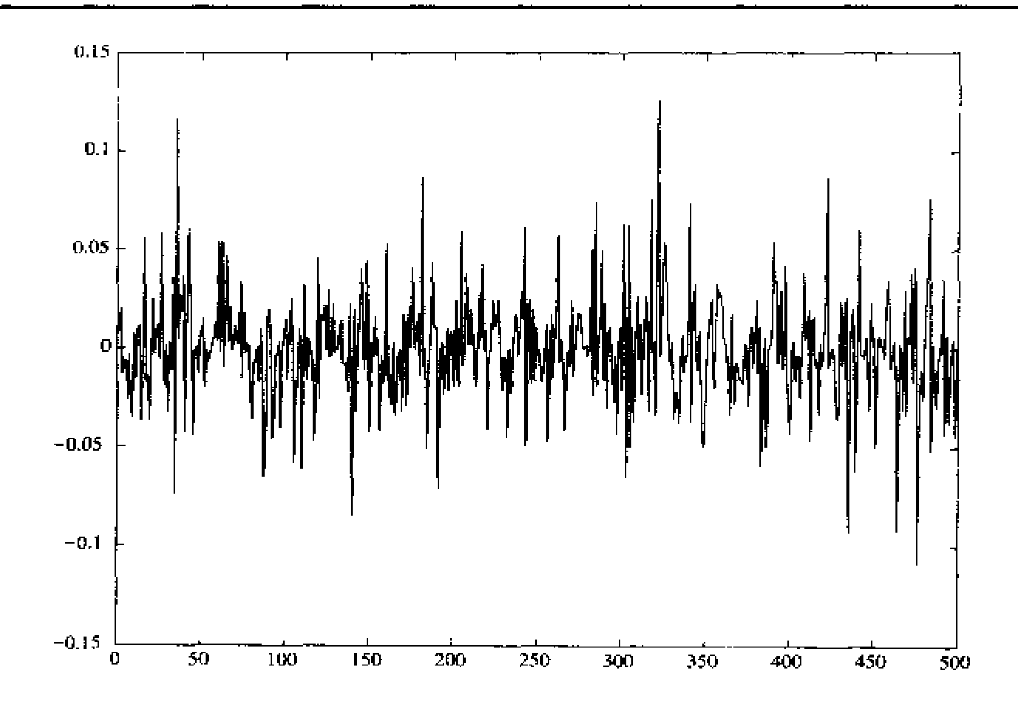


图 4.16　64 量化电平数 μ 律 PCM 的量化误差和量化器输入－输出关系



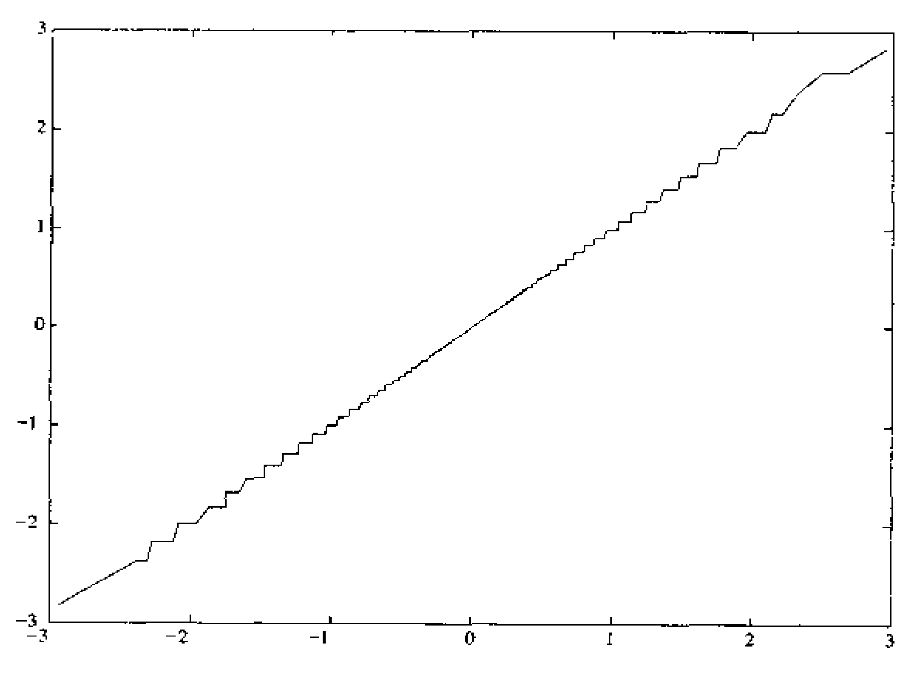


图 4.17 128 量化电平数 μ 律 PCM 的量化误差和量化器输入－输出关系

题 解

用下面的 MATLAB 指令产生所要求的序列

$$a = [20 * \text{randn}(1,20)\text{randn}(1,480)]$$

现在就可以用文件 u _ pcm.m 和 mula _ pcm.m 求出 SQNR。所得的 SQNR 分别是 20.49 dB 和 24.95 dB。在这种情况下,非均匀 PCM 的性能肯定优于均匀 PCM 的性能。

4.4　习题

4.1　为概率是

$$p=\{0.1,0.05,0.21,0.07,0.02,0.2,0.2,0.15\}$$

的信息源设计一个 Huffman 码。通过计算平均码字长度和信源的熵求这个码的效率。

4.2　由概率向量 $p=\{0.2,0.3,0.1,0.4\}$ 描述的一个离散无记忆信息源。

a. 写一个 MATLAB 文件计算该信源 K 次延伸的概率，K 已知。

b. 为这个信源和它的 K 次延伸($K=1,2,3,4,5$)设计 Huffman 码。

c. 画出平均码字长度(每信源输出)作为 K 的函数的图。

4.3　表 4.1 给出了出现在书面英语中的字母的概率。

a. 求书面英语的熵。

b. 为书面英语设计一个 Huffman 码。

c. 求平均码字长度和该 Huffman 码的效率。

4.4　对概率向量 $p=\{0.5,0.25,0.125,0.125\}$ 的离散无记忆信源重做习题 4.2。试解释为什么这时的结果不同于习题 4.2 所得的结果。

4.5　某二元信源有两个输出 a_1 和 a_2，其概率分别为 0.95 和 0.05。

a. 为该信息源及其 n 次延拓(即每次取 n 个字符)，$n=2,3,4,5,6,7$，设计 Huffman 码，并对每种情况求**单个信源输出**的平均码字长度。

b. 作为 n 的函数，画出在 a 步骤中求得的每单个信源输出的平均码字长度。

c. 对概率分别为 0.6 和 0.4 的二元信源重做 a 和 b 步骤，并注意与第一种信源的差别。

4.6　有一个零均值、单位方差的高斯分布连续信息源，用均匀对称量化器量化，量化区域的长度是 1，量化电平数是 N。对 $N=2,3,4,5,6,7$ 和 8，求已量化信源输出的熵，并画出它作为 N 的函数的图。在同一个图上，画出 $\log_2 N \sim N$ 的图，并解释为什么这两条曲线不同。

4.7　用一个均匀量化器对零均值、单位方差的高斯源进行量化，这个量化器在区间[－10，10]内非均匀量化。假定量化电平设在各量化区域的中间点，求出并画出当 $N=3,4,5,6,7,8,9$ 和 10 时，均方失真作为量化电平数 N 的函数的图。

4.8　考虑两个随机变量 X_1 和 X_2，其中 X_1 是均值为零、方差为 2 的高斯随机变量，而 X_2 则是概率密度函数

表 4.1　书面英语中的字母的概率

字母	概率
A	0.0642
B	0.0127
C	0.0218
D	0.0317
E	0.1031
F	0.0208
G	0.0152
H	0.0467
I	0.0575
J	0.0008
K	0.0049
L	0.0321
M	0.0198
N	0.0574
O	0.0632
P	0.0152
Q	0.0008
R	0.0484
S	0.0514
T	0.0796
U	0.0228
V	0.0083
W	0.0175
X	0.0013
Y	0.0164
Z	0.0005
字的空间	0.1859

$$f(x_2)=\frac{1}{2}e^{-|x_2|}$$

的拉普拉斯随机变量。

a. 证明这两个随机变量有相等的方差。

b. 利用 lloydmax.m 文件对这两个信息源设计量化电平为 2,3,…,10 的量化器,求出对应的失真,并以列表形式给出。

c. 对每个随机变量 X_i,定义 SQNR 为 $\mathrm{SQN}_i=\frac{E(X_i^2)}{D_i}$, $i=1,2$,并对不同的 $N(2\sim10)$值在同一个图上作为 N 的函数画出 SQNR_1 和 SQNR_2。

4.9 在习题 4.7 所画的同一个图上,画出当量化电平取在各量化区域的质心时的均方失真的图。当 N 为何值时这两个图很接近。为什么?

4.10 为一个零均值、单位方差的高斯源设计最优非均匀量化器,其量化电平数为 $N=2,3,4,5,6,7$ 和 8,对每种情况求已量化信源的熵 $H(\hat{X})$,为该信源设计的 Huffman 码的平均码字长度为 R,在同一个图上画出作为 N 的函数的 $H(\hat{X})$, R 和 $\log_2 N$ 的图。

4.11 概率密度函数为

$$f(x)=\frac{\lambda}{2}e^{-\lambda|x|}$$

的随机变量定义为拉普拉斯随机变量,式中 $\lambda>0$ 是某个给定常数。

a. 证明拉普拉斯随机变量的方差等于 $2/\lambda^2$。

b. 假设 $\lambda=1$,为这个信源设计量化电平数 $N=2,3,4,5,6,7$ 和 8 的均匀量化器。按惯例取有用区间$[-10\sigma,10\sigma]$, σ 是该信源的标准方差。

c. 在同一个图上,画出作为 N 的函数的已量化源的熵和 $\log_2 N$ 的图。

4.12 用最优非均匀量化器代替均匀量化器重做习题 4.11。

4.13 为拉普拉斯源设计一个最优 8 电平量化器,λ 在区间[0.1,5]内变化,画出作为 λ 的函数的所得均方失真的图。

4.14 对习题 4.11 中的拉普拉斯源,$\lambda=\sqrt{2}$,用 $N=2,3,4,5,6,7$ 和 8 设计最优非均匀量化器(注意,λ 的这一选取会导致一个零均值和单位方差的拉普拉斯源)。对该信源画出作为 N 的函数的均方误差的图。将这些结果与量化一个零均值和单位方差的高斯源所得的结果进行比较。

4.15 用 $\rho=0.1,0.3,0.7,0.99$ 做解说题 4.10,对结果进行比较并讨论 ρ 对失真的影响。

4.16 利用标量量化器解习题 4.15,将结果与习题 4.15 的结果进行比较,并讨论存储如何影响一个矢量量化器的性能。

4.17 周期信号 $x(t)$的周期为 2,在区间[0,2]内定义为:

$$x(t)=\begin{cases}t, & 0\leqslant t\leqslant 1\\ -t+2, & 1\leqslant t\leqslant 2\end{cases}$$

a. 对这个信号设计一个 8 电平的均匀 PCM 量化器,画出这个系统的已量化输出。

b. 画出该系统的量化误差。

c. 通过计算误差信号的功率,求该系统的 SQNR(以 dB 计)。

d. 用 16 电平均匀 PCM 系统重做 a,b 和 c 这 3 个步骤。

4.18　产生 1000 个零均值和方差为 1 的高斯序列，为这个序列设计 4,8,16,32 和 64 电平的均匀 PCM，作为分配给每个信源输出的比特数的函数画出所给 SQNR（以 dB 计）的图。

4.19　产生 1000 个零均值和单位方差的高斯序列，用 6 比特/符号的均匀 PCM 方法量化它，产生的 6000 比特经由一个噪声信道发送到接收端，信道的差错概率是 p，当 $p=10^{-3}, 5\times10^{-3}, 10^{-2}, 5\times10^{-2}$、0.1 和 0.2 时，作为 p 的函数画出总的 SQNR（以 dB 计）的图。为了仿真噪声的效果，可以产生具有这些概率的二进制随机序列，再将它们加（按模 2）到这个已编码的序列上。

4.20　用 $\mu=255$ 的非均匀 μ 律 PCM 重做习题 4.17。

4.21　用 $\mu=255$ 的非均匀 μ 律 PCM 重做习题 4.18。

4.22　用 $\mu=255$ 的非均匀 μ 律 PCM 重做习题 4.19。

第 5 章　基带数字传输

5.1　概述

第 5 章讨论了经由加性高斯白噪声信道传输数字信息的几种基带数字调制和解调技术。讨论由二进制脉冲调制入手，然后介绍几种非二进制调制方法。对于这些不同的信号，我们给出了最佳接收机的概念，并通过平均差错概率对它们的性能做出评价。

5.2　二进制信号传输

在二进制通信系统中，由 0 和 1 的序列组成的二进制数据是用两种信号波形 $s_0(t)$ 和 $s_1(t)$ 来传输的，假设数据率是 R(bit/s)，那么每个比特就按照规则：

$$0 \to s_0(t), \quad 0 \leqslant t \leqslant T_b$$
$$1 \to s_1(t), \quad 0 \leqslant t \leqslant T_b$$

映射为某个对应的信号波形，式中 $T_b = 1/R$ 定义为比特时间区间。假设数据比特流中的 0 和 1 是等概率的，即每个出现的概率都是 1/2，而且是互为统计独立的。

传输信号的信道假设被加性噪声 $n(t)$ 所污损，这样的信道称为**加性高斯白噪声（*AWGN*）信道**。$n(t)$ 就是功率谱为 $N_0/2$（W/Hz）的白色高斯过程的一个样本函数。这样，接收到的信号就可以表示成：

$$r(t) = s_i(t) + n(t), \quad i = 0,1 \quad 0 \leqslant t \leqslant T_b \tag{5.2.1}$$

接收机的任务就是在观察接收到的信号 $r(t)$ 之后，判断在区间 $0 \leqslant t \leqslant T_b$ 内发送的究竟是 0 还是 1，接收机总是要设计为使差错概率最小。这样的接收机称为**最佳接收机**。

5.2.1　AWGN 信道的最佳接收机

在几乎所有的数字通信方面的教材中，都指出对于 AWGN 信道的最佳接收机由两部分组成：一个是**信号相关器**或**匹配滤波器**；另一个是**检测器**。

信号相关器

信号相关器将接收到的信号 $r(t)$ 与两个可能的发送信号 $s_0(t)$ 和 $s_1(t)$ 做互相关，如图 5.1所示。也就是说，信号相关器计算在区间 $0 \leqslant t \leqslant T_b$ 内的如下两个输出：

$$r_0(t) = \int_0^t r(\tau) s_0(\tau) \mathrm{d}\tau$$
$$r_1(t) = \int_0^t r(\tau) s_1(\tau) \mathrm{d}\tau \tag{5.2.2}$$

在 $t = T_b$ 时对这两个输出采样，并将已采样输出馈给检测器。

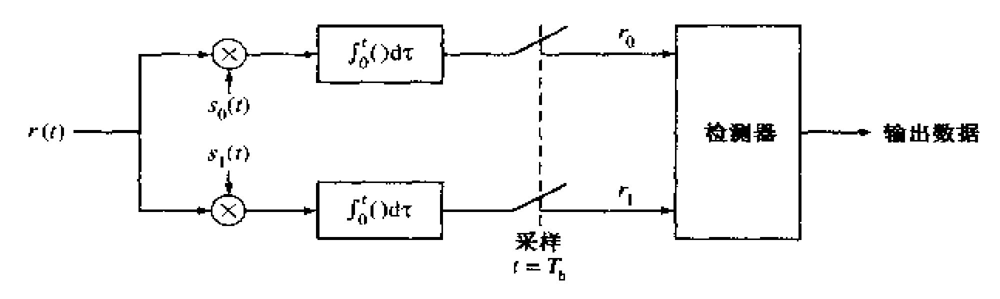


图 5.1　接收信号 $r(t)$ 与两个发送信号的互相关

解说题

解说题 5.1　[信号相关器]

假设信号波形 $s_0(t)$ 和 $s_1(t)$ 如图 5.2 所示，并设 $s_0(t)$ 是已发送信号，那么已接收信号是：

$$r(t) = s_0(t) + n(t), \qquad 0 \leqslant t \leqslant T_b \tag{5.2.3}$$

求在采样瞬时的相关器的输出。

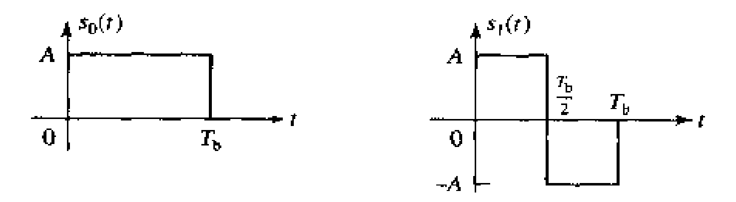


图 5.2　二进制通信系统的信号波形 $s_0(t)$ 和 $s_1(t)$

题　解

当信号 $r(t)$ 用图 5.1 所示的两个信号相关器处理时，在采样瞬时 $t = T_b$ 的输出 r_0 和 r_1 为：

$$\begin{aligned} r_0 &= \int_0^{T_b} r(t)s_0(t)dt \\ &= \int_0^{T_b} s_0^2(t)dt + \int_0^{T_b} n(t)s_0(t)dt \\ &= \mathscr{E} + n_0 \end{aligned} \tag{5.2.4}$$

和

$$\begin{aligned} r_1 &= \int_0^{T_b} r(t)s_1(t)dt \\ &= \int_0^{T_b} s_0(t)s_1(t)dt + \int_0^{T_b} n(t)s_1(t)dt \\ &= n_1 \end{aligned} \tag{5.2.5}$$

其中，n_0 和 n_1 都是在信号相关器输出端的噪声分量，即

$$\begin{aligned} n_0 &= \int_0^{T_b} n(t)s_0(t)dt \\ n_1 &= \int_0^{T_b} n(t)s_1(t)dt \end{aligned} \tag{5.2.6}$$

而 $\mathscr{E} = A^2 T_b$ 是信号 $s_0(t)$ 和 $s_1(t)$ 的能量。另外还要注意到，这两个信号波形是正交的，即

$$\int_0^{T_b} s_0(t)s_1(t)\mathrm{d}t = 0 \tag{5.2.7}$$

另一方面，当 $s_1(t)$ 是已发送信号时，已接收信号是：

$$r(t) = s_1(t) + n(t) \qquad 0 \leqslant t \leqslant T_b$$

很容易证明，在这种情况下信号相关器的输出是：

$$\begin{aligned} r_0 &= n_0 \\ r_1 &= \mathscr{E} + n_1 \end{aligned} \tag{5.2.8}$$

图5.3给出了当发送信号 $s_0(t)$ 和 $s_1(t)$ 时在区间 $0 \leqslant t \leqslant T_b$ 内两个无噪声相关器的输出。

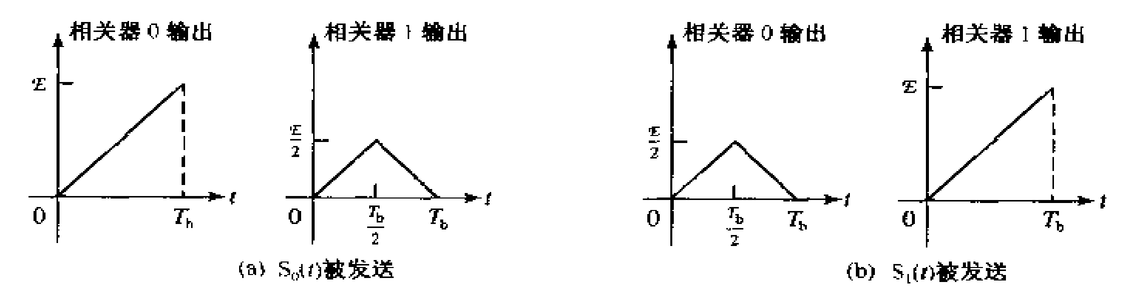


图5.3　无噪声相关器的输出

因为 $n(t)$ 是功率谱为 $N_0/2$ 的白色高斯过程的一个样本函数，所以噪声分量 n_0 和 n_1 就是零均值高斯型的，即

$$\begin{aligned} E(n_0) &= \int_0^{T_b} s_0(t)E[n(t)]\mathrm{d}t = 0 \\ E(n_1) &= \int_0^{T_b} s_1(t)E[n(t)]\mathrm{d}t = 0 \end{aligned} \tag{5.2.9}$$

并且方差 $\sigma_i^2(i=1,2)$ 为：

$$\begin{aligned} \sigma_i^2 &= E(n_i^2) \\ &= \int_0^{T_b}\int_0^{T_b} s_i(t)s_i(\tau)E[n(t)n(\tau)]\mathrm{d}t\mathrm{d}\tau \\ &= \frac{N_0}{2}\int_0^{T_b} s_i(t)s_i(\tau)\delta(t-\tau)\mathrm{d}t\mathrm{d}\tau \\ &= \frac{N_0}{2}\int_0^{T_b} s_i^2(t)\mathrm{d}t \qquad (5.2.10) \\ &= \frac{\mathscr{E}N_0}{2}, \quad i = 0,1 \qquad (5.2.11) \end{aligned}$$

因此，当 $s_0(t)$ 被发送时，r_0 和 r_1 的概率密度函数是：

$$\begin{aligned} p(r_0 \mid s_0(t)\ \text{被发送}) &= \frac{1}{\sqrt{2\pi}\sigma}\mathrm{e}^{-(r_0-\mathscr{E})^2/2\sigma^2} \\ p(r_1 \mid s_0(t)\ \text{被发送}) &= \frac{1}{\sqrt{2\pi}\sigma}\mathrm{e}^{-r_1^2/2\sigma^2} \end{aligned} \tag{5.2.12}$$

这两个概率密度函数 $p(r_0|0)$ 和 $p(r_1|0)$ 如图5.4所示。类似地，当 $s_1(t)$ 被发送时，r_0 就是零均值和方差为 σ^2 的高斯随机变量，而 r_1 则是均值为$\mathscr{E}$和方差为 σ^2 的高斯随机变量。

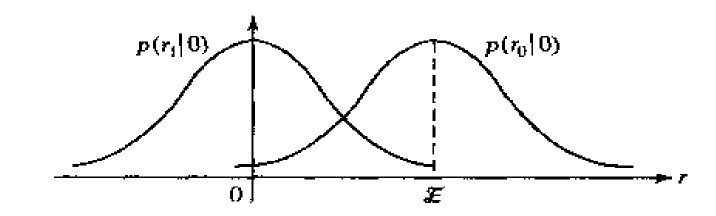


图 5.4　当 $s_0(t)$被发送时的概率密度函数 $p(r_0|0)$和 $p(r_1|0)$

匹配滤波器

匹配滤波器为解调已接收到信号 $r(t)$的信号相关器提供了另一种方法。对于信号波形 $s(t), 0 \leqslant t \leqslant T_b$，匹配的滤波器的冲激响应为：

$$h(t) = s(T_b - t), \qquad 0 \leqslant t \leqslant T_b \tag{5.2.13}$$

这样，当输入波形是 $s(t)$时，在匹配滤波器输出端的信号波形 $y(t)$由下面的卷积积分给出：

$$y(t) = \int_0^t s(\tau)h(t-\tau)\mathrm{d}\tau \tag{5.2.14}$$

如果将式(5.2.13)中的 $h(t-\tau)$代入式(5.2.14)中，可得：

$$y(t) = \int_0^t s(\tau)s(T_b - t + \tau)\mathrm{d}\tau \tag{5.2.15}$$

如果在 $t = T_b$ 时对 $y(t)$采样，可得：

$$y(T_b) = \int_0^{T_b} s^2(t)\mathrm{d}t = \mathscr{E} \tag{5.2.16}$$

其中，$\mathscr{E}$是信号 $s(t)$的能量。因此，匹配滤波器在采样瞬时 $t = T_b$ 的输出与信号相关器的输出是一样的。

解说题

解说题 5.2　[匹配滤波器]

考虑用匹配滤波器对图 5.2 所示的两个信号波形解调，并求输出。

题　解

两个匹配滤波器的冲激响应是：

$$\begin{aligned} h_0(t) &= s_0(T_b - t) \\ h_1(t) &= s_1(T_b - t) \end{aligned} \tag{5.2.17}$$

如图 5.5 所示。注意，将信号 $s(t)$反转得到 $s(-t)$，然后将反转信号 $s(-t)$延迟 T_b 就得出 $s(T_b - t)$。

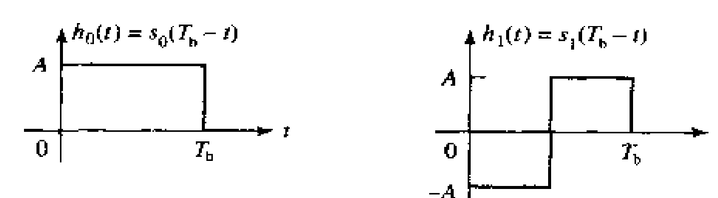


图 5.5　对信号 $s_0(t)$和 $s_1(t)$的匹配滤波器的冲激响应

现在假定信号波形 $s_0(t)$ 被发送，那么已接收信号 $r(t)=s_0(t)+n(t)$ 通过这两个匹配滤波器。冲激响应为 $h_0(t)$ 的滤波器对信号分量 $s_0(t)$ 的响应如图5.6(a)所示。冲激响应为 $h_1(t)$ 的滤波器对信号分量 $s_0(t)$ 的响应如图5.6(b)所示。所以，在采样瞬时 $t=T_b$，冲激响应为 $h_0(t)$ 和 $h_1(t)$ 的两个匹配滤波器的输出分别是：

$$r_0=\mathscr{E}+n_0$$
$$r_1=n_1 \tag{5.2.18}$$

注意，这些输出与在 $t=T_b$ 时对信号相关器采样所得的输出是相同的。

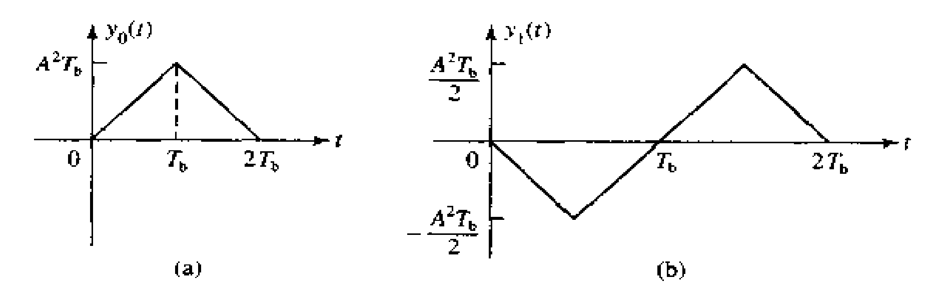


图5.6　当 $s_0(t)$ 被发送时匹配滤波器的信号输出

检测器

检测器观察到相关器或匹配滤波器的输出 r_0 和 r_1，并判决所发送的信号波形是 $s_0(t)$ 还是 $s_1(t)$，这就分别相应于传输的是一个0或1。**最佳检测器**就是使差错概率最小的检测器。

解说题

解说题5.3　[二进制检测]

现考虑图5.2所示的信号的检测器，这些信号是等概率的并有相等的能量。这两个信号的最佳检测器将比较 r_0 和 r_1，并判断出：当 $r_0>r_1$ 时传输的是0，当 $r_1>r_0$ 时传输的是1。求差错概率。

题　解

当 $s_0(t)$ 是发送的信号波形时，差错概率是：

$$P_e=P(r_1>r_0)=P(n_1>\mathscr{E}+n_0)=P(n_1-n_0>\mathscr{E}) \tag{5.2.19}$$

因为 n_1 和 n_0 是零均值高斯随机变量，它们的差 $x\equiv n_1-n_0$ 也是零均值高斯型的，随机变量 x 的方差是：

$$E(x^2)=E[(n_1-n_0)^2]=E(n_1^2)+E(n_0^2)-2E(n_1n_0) \tag{5.2.20}$$

但是 $E(n_1n_0)=0$，因为这些信号波形是正交的，即

$$\begin{aligned}E(n_1n_0)&=E\int_0^{T_b}\int_0^{T_b}s_0(t)s_1(\tau)n(t)n(\tau)\mathrm{d}t\mathrm{d}\tau\\&=\frac{N_0}{2}\int_0^{T_b}\int_0^{T_b}s_0(t)s_1(\tau)\delta(t-\tau)\mathrm{d}t\mathrm{d}\tau\\&=\frac{N_0}{2}\int_0^{T_b}s_0(t)s_1(t)\mathrm{d}t\\&=0\end{aligned} \tag{5.2.21}$$

因此

$$E(x^2) = 2\left(\frac{\mathscr{E}N_0}{2}\right) = \mathscr{E}N_0 \equiv \sigma_x^2 \tag{5.2.22}$$

所以，差错概率是：

$$\begin{aligned} P_e &= \frac{1}{\sqrt{2\pi}\sigma_x}\int_{\mathscr{E}}^{\infty} e^{-x^2/2\sigma_x^2}\,dx \\ &= \frac{1}{\sqrt{2\pi}}\int_{\sqrt{\mathscr{E}/N_0}}^{\infty} e^{-x^2/2}\,dx \\ &= Q\left(\sqrt{\frac{\mathscr{E}}{N_0}}\right) \end{aligned} \tag{5.2.23}$$

比值$\mathscr{E}/N_0$ 称为信噪比(SNR)。

本例所给出的检测器性能的导出是基于传送的信号波形为 $s_0(t)$的。读者可以证明，当传送的是 $s_1(t)$时得出的差错概率与传送 $s_0(t)$时得出的差错概率相同。因为在这个数据序列中 0 和 1 是等概率的，所以平均差错概率就由式(5.2.23)给出。用下面给出的 MATLAB 脚本对这个表达式的差错概率求值，作为 SNR 的函数画在图 5.7 中。图中 SNR 是以对数标尺($10\log_{10}\mathscr{E}/N_0$)给出的。正如我们所预料的，差错概率随 SNR 的增加成指数律减小。

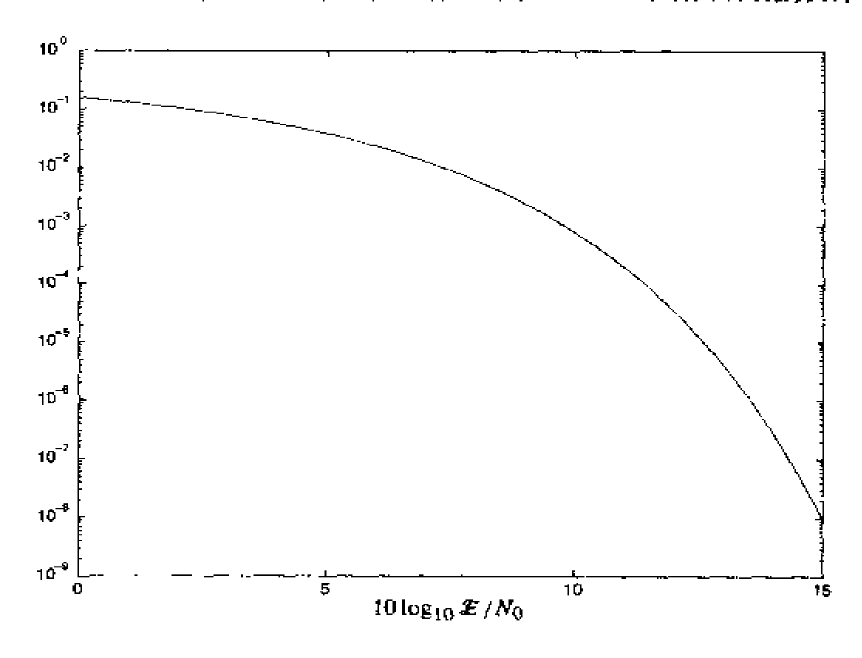


图 5.7　正交信号的差错概率

m 文件

```
% The MATLAB script that generates the probability of error versus the signal-to-noise ratio.
initial_snr=0;
final_snr=15;
snr_step=0.25;
snr_in_dB=initial_snr:snr_step:final_snr;
for i=1:length(snr_in_dB),
   snr=10^(snr_in_dB(i)/10);
   Pe(i)=Qfunct(sqrt(snr));
```

```
  echo off;
end;
echo on;
semilogy(snr_in_dB,Pe);
```

二进制通信系统的 Monte Carlo 仿真

在实际情况下,为了估计某个数字通信系统的差错概率,通常都用 Monte Carlo 计算机仿真来完成,尤其在对检测器的性能分析很困难时更是如此。现在对前述的二进制通信系统说明估计差错概率的方法。

解说题

解说题 5.4 [Monte Carlo 仿真]

对一个使用相关器或匹配滤波器的二进制通信系统,用 Monte Carlo 仿真估计 P_e,并画出 P_e 与 SNR 的对比图。该系统的模型如图 5.8 所示。

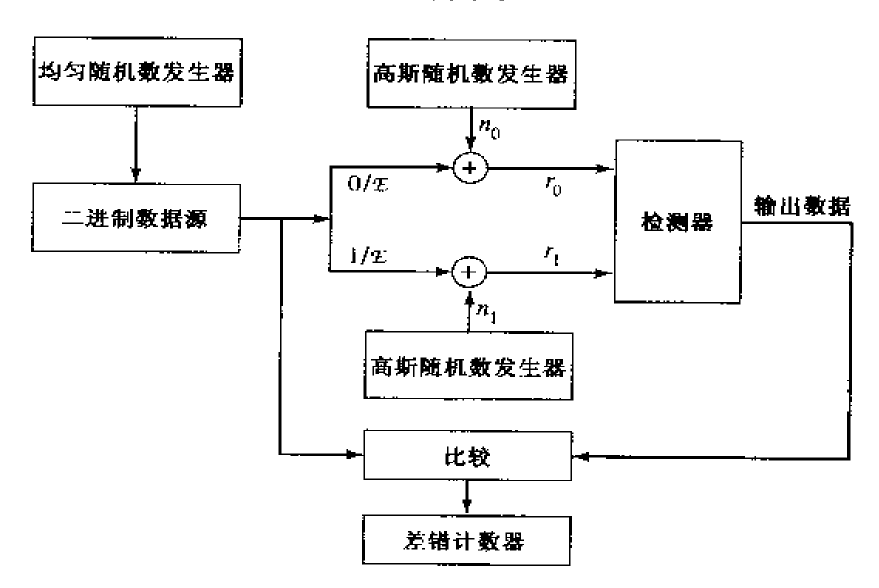


图 5.8 解说题 5.4 的仿真模型

题 解

我们先仿真产生随机变量 r_0 和 r_1,它们构成了检测器的输入。首先产生一个等概率出现并互为统计独立的二进制 0 和 1 的序列。为了实现这一点,使用一个产生范围在(0,1)内的均匀随机数的随机数发生器,如果产生的随机数在(0,0.5)以内,二进制源的输出就是 0,否则就是 1。若产生一个 0,那么 $r_0=\mathcal{E}+n_0, r_1=n_1$;若产生一个 1,那么 $r_0=n_0, r_1=\mathcal{E}+n_1$。

利用两个高斯噪声发生器产生加性噪声分量 n_0 和 n_1,它们的均值是零,方差是 $\sigma^2=\mathcal{E}\ N_0/2$。为了方便,可以将信号能量$\mathcal{E}$归一化到 1($\mathcal{E}=1$)而改变 σ^2。应该注意,这样 SNR(定义为$\mathcal{E}/N_0$)就等于 $1/(2\sigma^2)$。检测器输出与二进制发送序列进行比较,差错计数器用来计数比特差错数。

图 5.9 示意了在几个不同的 SNR 值下,传输 $N=10\ 000$ 个比特时的仿真结果。我们可以

看到仿真结果与由式(5.2.23)给出的理论值 P_e 之间的一致性。还应该注意到，$N = 10\ 000$ 个数据比特的仿真能够可靠地估计出差错概率在 $p_e = 10^{-3}$ 以下；换句话说，用 $N = 10\ 000$ 个数据比特，在对 p_e 的可靠估计下应该至少有 10 个差错。

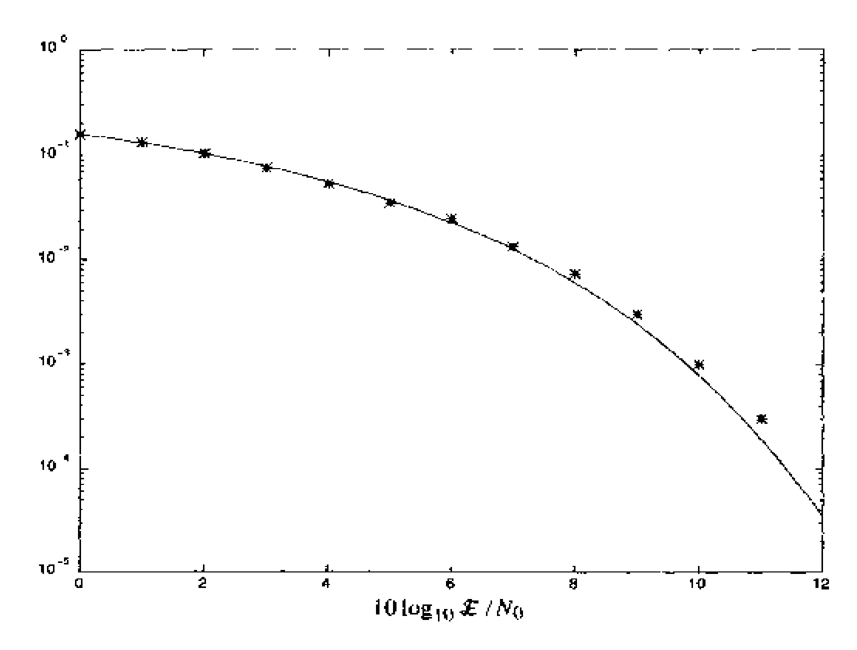


图 5.9　正交信号 Monte Carlo 仿真的差错概率与理论差错概率的比较

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 5.4.
echo on
SNRindB1=0:1:12;
SNRindB2=0:0.1:12;
for i=1:length(SNRindB1),
  % simulated error rate
  smld_err_prb(i)=smldPe54(SNRindB1(i));
  echo off ;
end;
echo on ;
for i=1:length(SNRindB2),
  SNR=exp(SNRindB2(i)*log(10)/10);
  % theoretical error rate
  theo_err_prb(i)=Qfunct(sqrt(SNR));
  echo off ;
end;
echo on;
% Plotting commands follow.
semilogy(SNRindB1,smld_err_prb,'*');
hold
semilogy(SNRindB2,theo_err_prb);
```

m 文件

```
function [p]=smldPe54(snr_in_dB)
% [p]=smldPe54(snr_in_dB)
%                SMLDPE54  finds the probability of error for the given
%                snr_in_dB, signal-to-noise ratio in dB.
E=1;
SNR=exp(snr_in_dB*log(10)/10);          % signal-to-noise ratio
sgma=E/sqrt(2*SNR);                     % sigma, standard deviation of noise
N=10000;
% generation of the binary data source
for i=1:N,
  temp=rand;                            % a uniform random variable over (0,1)
  if (temp<0.5),
    dsource(i)=0;                       % With probability 1/2, source output is 0.
  else
    dsource(i)=1;                       % With probability 1/2, source output is 1.
  end
end;
% detection, and probability of error calculation
numoferr=0;
for i=1:N,
  % matched filter outputs
  if (dsource(i)==0),
    r0=E+gngauss(sgma);
    r1=gngauss(sgma);                   % if the source output is "0"
  else
    r0=gngauss(sgma);
    r1=E+gngauss(sgma);                 % if the source output is "1"
  end;
  % Detector follows.
  if (r0>r1),
    decis=0;                            % Decision is "0".
  else
    decis=1;                            % Decision is "1".
  end;
  if (decis~=dsource(i)),               % If it is an error, increase the error counter.
    numoferr=numoferr+1;
  end;
end;
p=numoferr/N;                           % probability of error estimate
```

思考题

在图 5.9 中,仿真和理论值在低信噪比下完全一致,而在高信噪比下一致性稍差,能解释为什么吗? 我们应该如何改变仿真过程,以使在高信噪比下也能得到更好的一致性?

5.2.2 其他二进制信号传输方法

上面所讨论的二进制信号传输方法基于利用正交信号,下面要讨论经由某个通信信道传输二进制信息的另外两种方法,一种使用反极性信号,另一种使用开关信号。

二进制信号传输的反极性信号

如果一种信号波形是另一种的负值，则认为这两个信号波形是**反极性的**(antipodal)。例如，图 5.10(a)给出了一对反极性信号，而图 5.10(b)给出了另一对反极性信号。

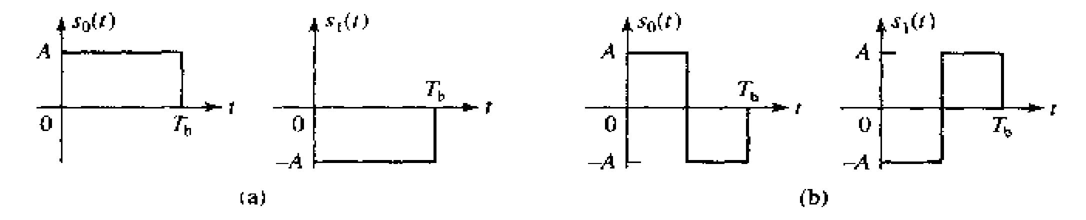
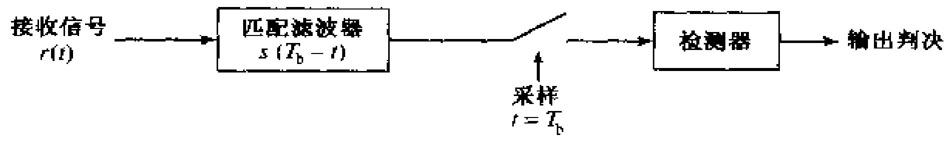


图 5.10　反极性信号对举例

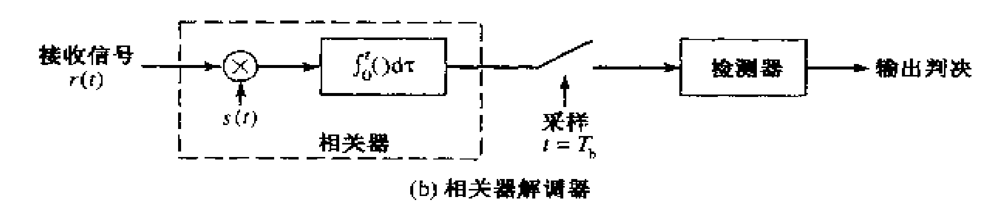
现在设想用反极性的信号波形 $s_0(t)=s(t)$ 和 $s_1(t)=-s(t)$ 来传输二进制信息。其中，$s(t)$ 是能量为 $\mathscr{E}$ 的某任意波形，从 AWGN 信道接收到的信号波形可表示为：

$$r(t)=\pm s(t)+n(t),\qquad 0\leqslant t\leqslant T_b \tag{5.2.24}$$

用于恢复该二进制信息的最佳接收机使用了一个相关器或与 $s(t)$ 匹配的匹配滤波器，再紧跟着一个检测器，如图 5.11 所示。



(a) 匹配滤波器解调器



(b) 相关器解调器

图 5.11　对于反极性信号的最佳接收机

假设发送的是 $s(t)$，接收到的信号就是：

$$r(t)=s(t)+n(t) \tag{5.2.25}$$

在采样瞬时 $t=T_b$，相关器或匹配滤波器的输出是：

$$r=\mathscr{E}+n \tag{5.2.26}$$

其中，$\mathscr{E}$ 是信号能量，n 是加性噪声分量，n 可以表示为：

$$n=\int_0^{T_b} n(t)s(t)\mathrm{d}t \tag{5.2.27}$$

因为加性噪声过程 $n(t)$ 是零均值的，因此有 $E(n)=0$。噪声分量 n 的方差是：

$$\begin{aligned}\sigma^2 &= E(n^2)\\ &= \int_0^{T_b}\int_0^{T_b} E[n(t)n(\tau)]s(t)s(\tau)\mathrm{d}t\mathrm{d}\tau\end{aligned}$$

$$= \frac{N_0}{2}\int_0^{T_b}\int_0^{T_b}\delta(t-\tau)s(t)s(\tau)\mathrm{d}t\mathrm{d}\tau$$
$$= \frac{N_0}{2}\int_0^{T_b}s^2(t)\mathrm{d}t$$
$$= \frac{N_0\mathscr{E}}{2} \tag{5.2.28}$$

结果,当发送 $s(t)$时,检测器的输入是:

$$p(r\mid s(t)\text{被发送}) \equiv p(r\mid 0) = \frac{1}{\sqrt{2\pi}\sigma}\mathrm{e}^{-(r-\mathscr{E})^2/(2\sigma^2)} \tag{5.2.29}$$

类似地,当发送信号波形 $-s(t)$时,检测器的输入是:

$$r = -\mathscr{E} + n \tag{5.2.30}$$

r 的概率密度函数是:

$$p(r\mid -s(t)\text{被发送}) \equiv p(r\mid 1) = \frac{1}{\sqrt{2\pi}\sigma}\mathrm{e}^{-(r+\mathscr{E})^2/(2\sigma^2)} \tag{5.2.31}$$

这两个概率密度函数如图 5.12 所示。

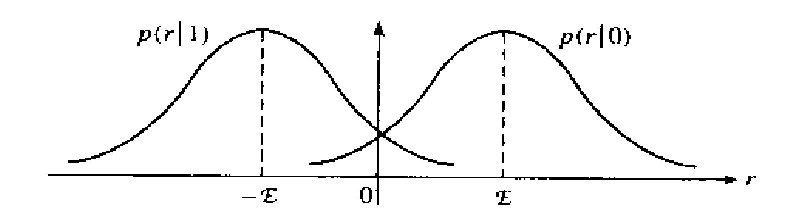


图 5.12 检测器输入的概率密度函数

对于等概率的信号波形,最佳检测器将 r 与阈值零相比较。若 $r>0$,则判决为 $s(t)$被发送;若 $r<0$,则判决为 $-s(t)$被发送。污损信号的噪声引起检测器中的差错。检测器的差错概率是很容易计算出来的。假定发送的是 $s(t)$,那么差错概率就等于 $r<0$ 的概率,即

$$P_e = P(r<0)$$
$$= \frac{1}{\sqrt{2\pi}\sigma}\int_{-\infty}^{0}\mathrm{e}^{-(r-\mathscr{E})^2/(2\sigma^2)}\mathrm{d}r$$
$$= \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{-\mathscr{E}/\sigma}\mathrm{e}^{-r^2/2}\mathrm{d}r$$
$$= Q\left(\frac{\mathscr{E}}{\sigma}\right)$$
$$= Q\left(\sqrt{\frac{2\mathscr{E}}{N_0}}\right) \tag{5.2.32}$$

当发送的是 $-s(t)$时,也能得出类似的结果。因此,当这两个信号波形是等概率的时,平均差错概率由式(5.2.32)给出。

当将反极性信号的差错概率与由式(5.2.32)给出的正交信号的差错概率比较时,可以看出,对同样的发送信号能量$\mathscr{E}$,反极性信号的结果有更好的性能。换一种方式说,在相同的性能(相同差错概率)下,反极性信号只需使用正交信号一半的发送能量,所以反极性信号比正交信号在效率上高出 3 dB。

解说题

解说题 5.5 [二进制反极性信号的仿真]

用 Monte Carlo 仿真估计并画出利用反极性信号的二进制通信系统的差错概率性能，系统的模型如图 5.13 所示。

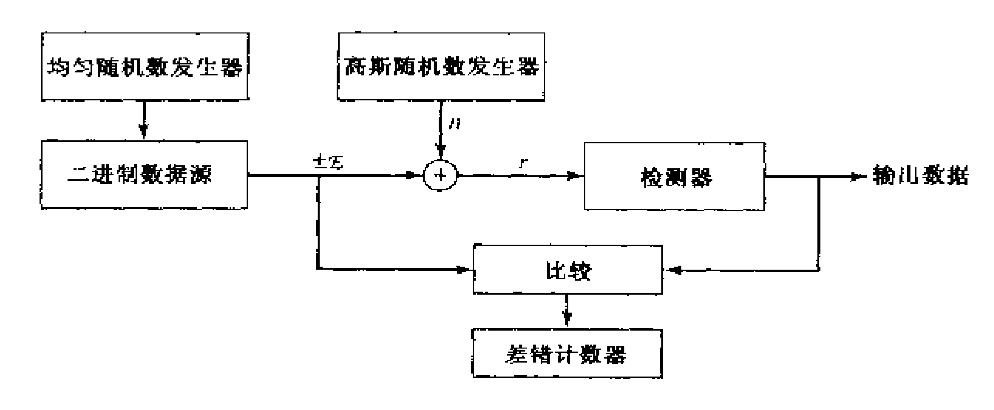


图 5.13 利用反极性信号的二进制通信系统模型

题 解

我们先仿真随机变量 r 的产生，它是检测器的输入。用一个均匀随机数发生器从二进制数据源中产生二进制信息序列。这个 0 和 1 的序列映射为 $\pm\mathcal{E}$ 的序列，其中 $\mathcal{E}$ 代表信号能量，可以归一化到 1。用一个高斯噪声发生器产生零均值、方差为 σ^2 的高斯随机数。检测器用阈值 0 与随机变量 r 进行比较。若 $r>0$，判决传送的比特是一个 0；若 $r<0$，判决传送的比特是一个 1。然后将检测器的输出与传输的信息比特的序列进行比较，比特差错被计数。图 5.14 说明了在几个不同的 SNR 值下，传输 $N=10\ 000$ 个比特时的仿真结果。由式(5.2.32)给出的 P_e 的理论值也画在图 5.14 中，以供比较。注意，对于 10 000 个传输比特，Monte Carlo 估计的 P_e 在 10^{-3} 以下不太准确。

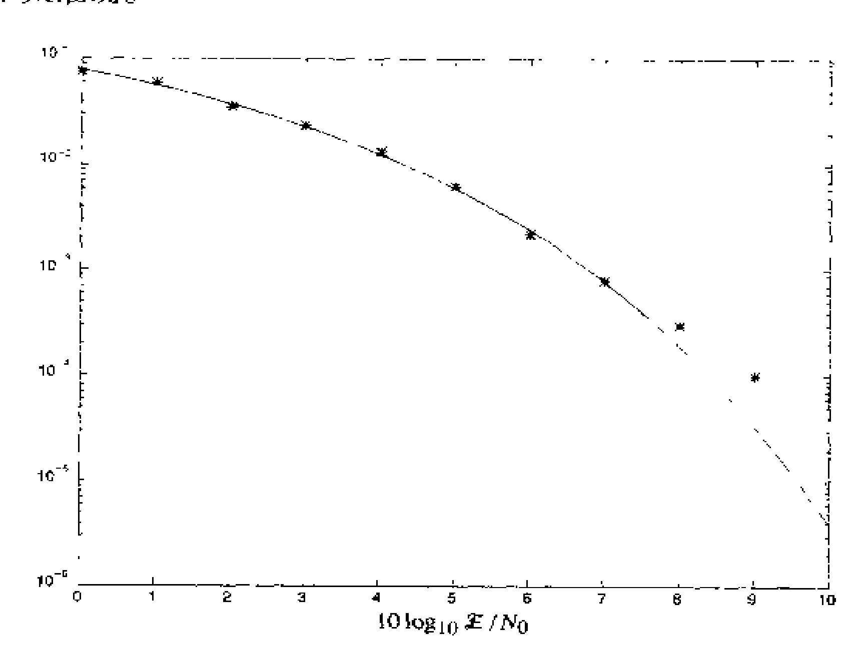


图 5.14 对于反极性信号，由 Monte Carlo 仿真的差错概率与理论差错概率的对比

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 5.5.
echo on
SNRindB1=0:1:10;
SNRindB2=0:0.1:10;
for i=1:length(SNRindB1),
  % simulated error rate
  smld_err_prb(i)=smldPe55(SNRindB1(i));
  echo off;
end;
echo on;
for i=1:length(SNRindB2),
  SNR=exp(SNRindB2(i)*log(10)/10);
  % theoretical error rate
  theo_err_prb(i)=Qfunct(sqrt(2*SNR));
  echo off;
end;
echo on;
% Plotting commands follow.
semilogy(SNRindB1,smld_err_prb,'*');
hold
semilogy(SNRindB2,theo_err_prb);
```

m 文件

```
function [p]=smldPe55(snr_in_dB)
% [p]=smldPe55(snr_in_dB)
%             SMLDPE55  simulates the probability of error for the particular
%             value of snr_in_dB, signal-to-noise ratio in dB.
E=1;
SNR=exp(snr_in_dB*log(10)/10);            % signal-to-noise ratio
sgma=E/sqrt(2*SNR);                       % sigma, standard deviation of noise
N=10000;
% Generation of the binary data source follows.
for i=1:N,
   temp=rand;                             % a uniform random variable over (0,1)
   if (temp<0.5),
      dsource(i)=0;                       % With probability 1/2, source output is 0.
   else
      dsource(i)=1;                       % With probability 1/2, source output is 1.
   end
end;
% The detection, and probability of error calculation follows.
numoferr=0;
for i=1:N,
   % the matched filter outputs
   if (dsource(i)==0),
      r=-E+gngauss(sgma);                 % if the source output is "0"
   else
      r=E+gngauss(sgma);                  % if the source output is "1"
   end;
   % Detector follows.
   if (r<0),
      decis=0;                            % Decision is "0".
```

```
    else
        decis=1;                        % Decision is "1".
    end;
    if (decis~=dsource(i)),             % If it is an error, increase the error counter.
        numoferr=numoferr+1;
    end;
end;
p=numoferr/N;                           % probability of error estimate
```

二进制信号传输的开关信号

二进制信息序列也可以用开关信号来传送。为了发送一个 0，在持续期为 T_b 的时间区间内不传送任何信号；为了发送一个 1，传送了信号波形 $s(t)$。结果，接收到的信号波形可表示为：

$$r(t)=\begin{cases}n(t), & \text{若 0 被传输}\\ s(t)+n(t), & \text{若 1 被传输}\end{cases}\tag{5.2.33}$$

其中，$n(t)$代表加性高斯白噪声。

与反极性信号的情况一样，最佳接收机由一个相关器或与 $s(t)$匹配的匹配滤波器（它的输出在 $t=T_b$ 被采样），再紧跟着一个检测器组成，它将采样输出与阈值 α 进行比较。若 $r>\alpha$，则声明传输的是 1，否则就声明传输的是 0。

检测器的输入可以表示为：

$$r=\begin{cases}n, & \text{若 0 被传输}\\ \mathcal{E}+n, & \text{若 1 被传输}\end{cases}\tag{5.2.34}$$

其中，n 是零均值、方差 $\sigma^2=\mathcal{E}N_0/2$ 的高斯随机变量。因此，随机变量 r 的条件概率密度函数是：

$$p(r\mid 0)=\frac{1}{\sqrt{2\pi}\sigma}\mathrm{e}^{-r^2/(2\sigma^2)},\qquad \text{若 0 被传输}$$

$$p(r\mid 1)=\frac{1}{\sqrt{2\pi}\sigma}\mathrm{e}^{-(r-\mathcal{E})^2(2\sigma^2)},\quad \text{若 1 被传输}$$

这些概率密度函数如图 5.15 所示。

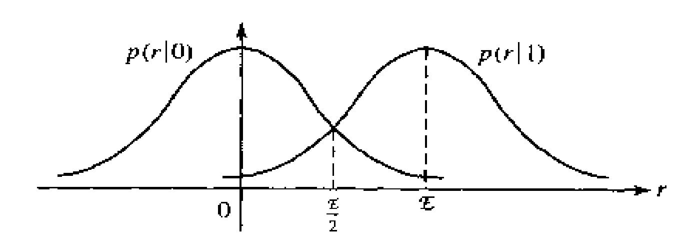


图 5.15　对于开关信号，在相关器输出端的已接收信号的概率密度函数

当传输的是 0 时，差错概率为：

$$P_{e0}(\alpha)=P(r>\alpha)=\frac{1}{\sqrt{2\pi}\sigma}\int_{\alpha}^{\infty}\mathrm{e}^{-r^2/(2\sigma^2)}\mathrm{d}r\tag{5.2.35}$$

其中，α 是阈值。另一方面，当传输的是 1 时，差错概率是：

$$P_{e1}(\alpha) = P(r < \alpha) = \frac{1}{\sqrt{2\pi}\sigma}\int_{-\infty}^{\infty} e^{-(r-\mathscr{E})^2/(2\sigma^2)}\,dr \tag{5.2.36}$$

假定二进制信息比特是等概率的,其平均差错概率为:

$$P_e(\alpha) = \frac{1}{2}P_{e0}(\alpha) + \frac{1}{2}P_{e1}(\alpha) \tag{5.2.37}$$

使这个平均差错概率最小的值 α 可以通过对 $P_e(\alpha)$ 微分并解出最佳阈值而求得。很容易证明,这个最佳阈值是:

$$\alpha_{opt} = \frac{\mathscr{E}}{2} \tag{5.2.38}$$

将这个最佳值代入式(5.2.35),式(5.2.36)和式(5.2.37),可得出差错概率是:

$$P_e(\alpha_{opt}) = Q\left(\sqrt{\frac{\mathscr{E}}{2N_0}}\right) \tag{5.2.39}$$

可以看出,使用开关信号时误码率的性能不如反极性信号时那么好。似乎与反极性信号相比差 6 dB,与正交信号相比差 3 dB。然而,对开关信号而言,其平均发送的能量比反极性信号都少 3 dB。因此,当与其他信号类型进行性能比较时,这个差别也应该计入。

解说题

解说题 5.6 [开关信号仿真]

用 Monte Carlo 仿真估计并画出使用开关信号的二进制通信系统的性能。

题解

要仿真的这个系统的模型与图 5.13 所示的类似,除了一个信号是 0 以外。因此,按式(5.2.34)产生一个随机变量 $\{r_i\}$ 序列,检测器将该随机变量 $\{r_i\}$ 与最佳阈值 $\mathscr{E}/2$ 进行比较,并做出适当的判决。图 5.16 给出了基于 10 000 个二进制数字的估计差错概率,由式(5.2.39)给出的理论上的误码率也示于图 5.16。

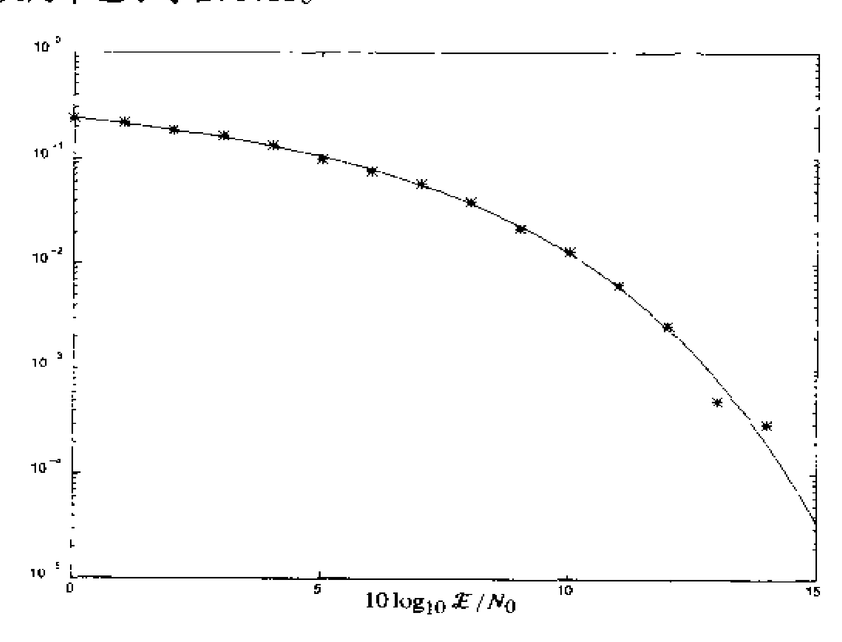


图 5.16 对于开关信号,Monte Carlo 仿真的差错概率与理论差错概率的比较

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 5.6.
echo on
SNRindB1=0:1:15;
SNRindB2=0:0.1:15;
for i=1:length(SNRindB1),
   smld_err_prb(i)=smldPe56(SNRindB1(i));          % simulated error rate
   echo off;
end;
echo on;
for i=1:length(SNRindB2),
   SNR=exp(SNRindB2(i)*log(10)/10);                % signal-to-noise ratio
   theo_err_prb(i)=Qfunct(sqrt(SNR/2));            % theoretical error rate
   echo off;
end;
echo on;
% Plotting commands follow.
semilogy(SNRindB1,smld_err_prb,'*');
hold
semilogy(SNRindB2,theo_err_prb);
```

m 文件

```
function [p]=smldPe56(snr_in_dB)
% [p]=smldPe56(snr_in_dB)
%              SMLDPE56 simulates the probability of error for a given
%              snr_in_dB, signal-to-noise ratio in dB.

E=1;
alpha_opt=1/2;
SNR=exp(snr_in_dB*log(10)/10);              % signal-to-noise ratio
sgma=E/sqrt(2*SNR);                         % sigma, standard deviation of noise
N=10000;
% Generation of the binary data source follows.
for i=1:N,
   temp=rand;                               % a uniform random variable over (0,1)
   if (temp<0.5),
      dsource(i)=0;                         % With probability 1/2, source output is 0.
   else
      dsource(i)=1;                         % With probability 1/2, source output is 1.
   end
end;
% detection, and probability of error calculation
numoferr=0;
for i=1:N,
   % the matched filter outputs
   if (dsource(i)==0),
      r=gngauss(sgma);                      % if the source output is "0"
   else
      r=E+gngauss(sgma);                    % if the source output is "1"
   end;
   % Detector follows.
   if (r<alpha_opt),
```

```
    decis=0;                              % Decision is "0".
  else
    decis=1;                              % Decision is "1".
  end;
  if (decis~=dsource(i)),                 % If it is an error, increase the error counter.
    numoferr=numoferr+1;
  end;
end;
p=numoferr/N;                             % probability of error estimate
```

5.2.3 二进制信号的信号星座图

三种二进制信号(即反极性信号、开关信号和正交信号)都可以在几何上用"信号空间"中的点来表征。在反极性信号的情况下,信号是 $s(t)$和 $-s(t)$,每个都具有能量$\mathscr{E}$,这两个信号点落在实线上的 $\pm\sqrt{\mathscr{E}}$点处,如图 5.17(a)所示。反极性信号的一维几何表示之所以可能,是由于仅用一个信号波形或基函数,即 $s(t)$,就足以在信号空间内表示出这对反极性信号。开关信号也是一维信号,所以两个信号就落在实线上的 0 和$\sqrt{\mathscr{E}}$点处,如图 5.17(b)所示。

另一方面,二进制正交信号需要有一个二维的几何表示,因为有两个线性独立的函数 $s_0(t)$和 $s_1(t)$,它们构成了两种信号波形。这样,相应于这两个信号的信号点就在$(\sqrt{\mathscr{E}},0)$和$(0,\sqrt{\mathscr{E}})$上,如图 5.17(c)所示。

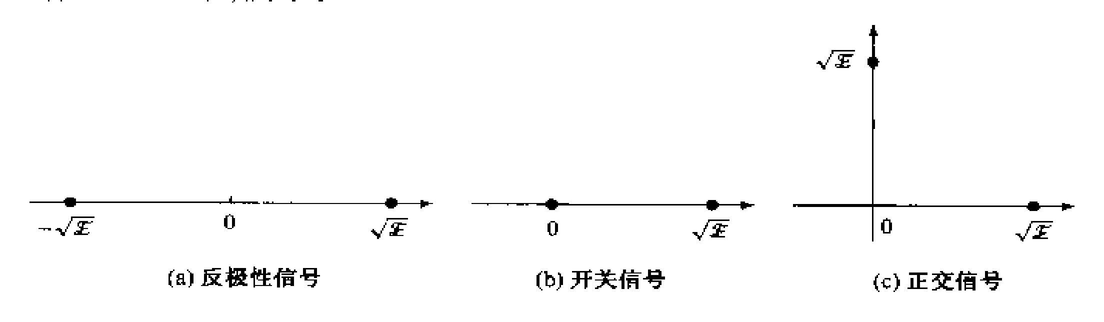


图 5.17 二进制信号的信号星座图

图 5.17 所示的二进制信号的几何表示就称为**信号星座图**。

解说题

解说题 5.7 [噪声影响和星座图]

噪声对二进制通信系统性能的影响可以从检测器输入端的已接收信号加噪声中看到。例如,考虑二进制正交信号,检测器的输入由一对随机变量(r_0,r_1)组成,这里的(r_0,r_1)或者为:

$$(r_0,r_1)=(\sqrt{\mathscr{E}}+n_0,n_1)$$

或者为:

$$(r_0,r_1)=(n_0,\sqrt{\mathscr{E}}+n_1)$$

噪声随机变量 n_0 和 n_1 都是零均值、方差为 σ^2 的独立高斯随机变量。和解说题 5.4 相同,分别按 $\sigma=0.1,\sigma=0.3$ 和 $\sigma=0.5$,用 Monte Carlo 仿真对每个 σ 值产生 100 个(r_0,r_1)样本。分别对每个 σ 值在不同的二维坐标图上画出这 100 个样本。信号能量$\mathscr{E}$可归一化到 1。

题　解

Monte Carlo 的仿真结果如图 5.18 所示。注意，在低噪声水平（小的 σ）下，噪声对通信系统的性能（误码率）影响是很小的。随着噪声功率的增加，噪声分量的大小增加了，引起了更多的误码。

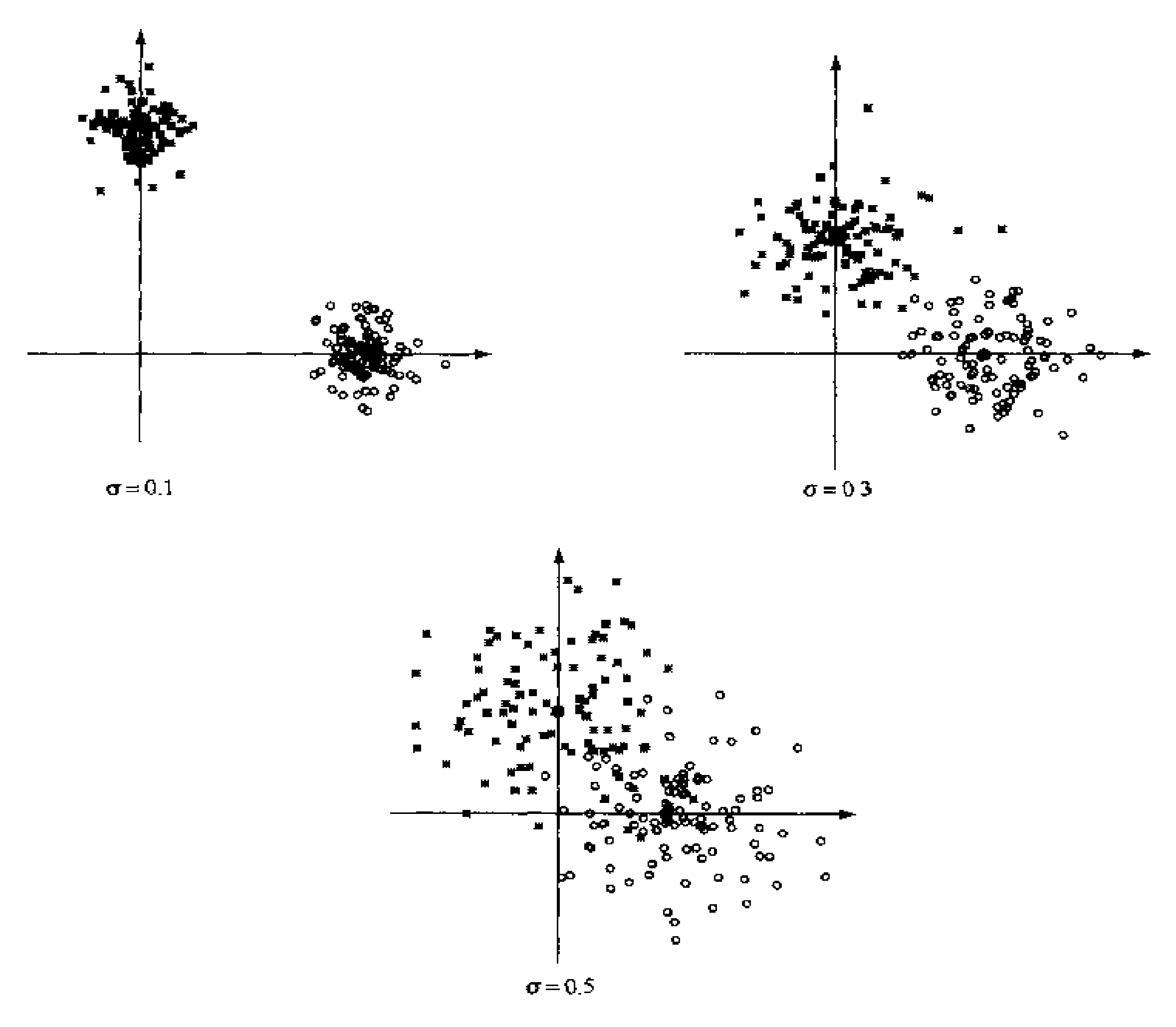


图 5.18　对于正交信号，在检测器输入端的接收信号点（Monte Carlo 仿真）

对于 $\sigma=0.5$，本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrated Problem 5.7.
echo on
n0=.5*randn(100,1);
n1=.5*randn(100,1);
n2=.5*randn(100,1);
n3=.5*randn(100,1);
x1=1.+n0;
y1=n1;
x2=n2;
y2=1.+n3;
plot(x1,y1,'o',x2,y2,'*')
axis('square')
```

5.3 多幅度信号传输

在前一节中,处理数字信息的传输用的都是二进制信号波形,因此每个信号波形仅传送一个比特信息。这一节要用取多个幅度电平的信号波形,因此每个信号波形就能传输多个比特的信息。

5.3.1 4幅度电平的信号波形

考虑一组信号波形形式为:

$$s_m(t) = A_m g(t), \qquad 0 \leqslant t \leqslant T \tag{5.3.1}$$

其中,A_m 是第 m 个波形的幅度,$g(t)$是矩形脉冲,定义为:

$$g(t) = \begin{cases} \sqrt{1/T}, & 0 \leqslant t \leqslant T \\ 0, & \text{其余 } t \end{cases} \tag{5.3.2}$$

其中,在脉冲 $g(t)$中的能量归一化到1。现在特别考虑这样一种情况:信号幅度取4种可能的等间隔值,即$\{A_m\} = \{-3d, -d, d, 3d\}$,或等效为:

$$A_m = (2m-3)d, \qquad m = 0,1,2,3 \tag{5.3.3}$$

其中,$2d$ 是两个相邻幅度电平之间的欧氏距离。4种信号波形如图5.19所示。我们称这组信号波形为**脉冲幅度调制**(PAM)信号。

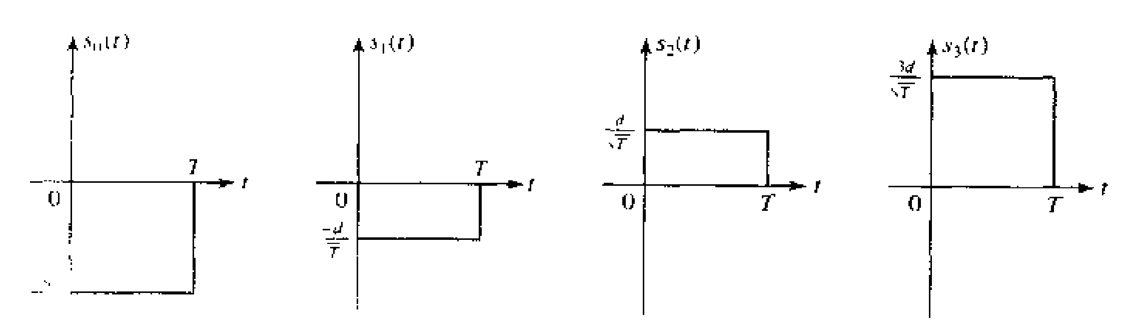


图5.19　多幅度信号波形

图5.19所示的4种PAM信号波形每个波形可用来传送2比特的信息,因此可把下面的信息比特对指定到这4种信号波形:

$$00 \rightarrow s_0(t)$$
$$01 \rightarrow s_1(t)$$
$$11 \rightarrow s_2(t)$$
$$10 \rightarrow s_3(t)$$

每个信息比特对{00,01,10,11}称为一个**符号**(symbol),时间持续期 T 称为**符号区间**(symbol interval)。注意,如果比特率是 $R = 1/T_b$,则符号区间就是 $T = 2T_b$。因为全部信号波形都是信号基函数 $g(t)$的幅度加权的结果,所以这些信号波形可以在几何上表示为实线上的一些点,因此这4种PAM信号的几何表示就是信号星座图,如图5.20所示。

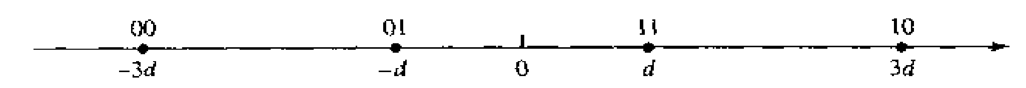


图 5.20 4 种 PAM 信号波形的信号星座图

与二进制信号的情况一样，假设 PAM 信号波形是经由 AWGN 信道传输的，接收到的信号即可表示为：

$$r(t) = s_i(t) + n(t), \quad i = 0,1,2,3, \quad 0 \leqslant t \leqslant T \tag{5.3.4}$$

其中，$n(t)$是功率谱 $N_0/2$(W/Hz)的高斯白噪声过程的一个样本函数。接收机的任务就是在观察到区间 $0 \leqslant t \leqslant T$ 的接收信号 $r(t)$之后，确定传输这 4 种信号波形中的哪一种。最佳接收机要设计成使符号差错概率最小。

5.3.2 AWGN 信道的最佳接收机

使差错概率最小的接收机可以将信号通过一个信号相关器或匹配滤波器，再紧跟着一个幅度检测器来实现。因为信号相关器和匹配滤波在采样瞬时都产生相同的输出，所以我们在讨论中仅考虑信号相关器。

信号相关器

信号相关器将接收到的信号 $r(t)$与信号脉冲 $g(t)$做互相关，并将它的输出在 $t = T$ 采样，因此信号相关器的输出是：

$$\begin{aligned} r &= \int_0^T r(t)g(t)\mathrm{d}t \\ &= \int_0^T A_i g^2(t)\mathrm{d}t + \int_0^T g(t)n(t)\mathrm{d}t \\ &= A_i + n \end{aligned} \tag{5.3.5}$$

式中 n 代表噪声分量，定义为：

$$n = \int_0^T g(t)n(t)\mathrm{d}t \tag{5.3.6}$$

注意，n 是一个高斯随机变量，其均值为：

$$E(n) = \int_0^T g(t)E[n(t)]\mathrm{d}t = 0 \tag{5.3.7}$$

方差为：

$$\begin{aligned} \sigma^2 &= E(n^2) \\ &= \int_0^T\int_0^T g(t)g(\tau)E[n(t)n(\tau)]\mathrm{d}t\mathrm{d}\tau \\ &= \frac{N_0}{2}\int_0^T\int_0^T g(t)g(\tau)\delta(t-\tau)\mathrm{d}t\mathrm{d}\tau \\ &= \frac{N_0}{2}\int_0^T g^2(t)\mathrm{d}t \\ &= \frac{N_0}{2} \end{aligned} \tag{5.3.8}$$

因此，信号相关器输出 r 的概率密度函数是：

$$p(r \mid s_i(t)\text{ 被传输}) = \frac{1}{\sqrt{2\pi}\sigma}\mathrm{e}^{-(r-A_i)^2/(2\sigma^2)} \tag{5.3.9}$$

其中,A_i 是 4 种可能幅度值之一。

检测器

检测器观察相关器的输出 r,并判决在信号区间内传输 4 种 PAM 信号中的哪一种。在下面的关于最佳检测器的性能讨论中,假设这 4 种可能的幅度电平是等概率的。

正如在图 5.20 的信号星座图中所表明的,因为接收到的信号幅度 A_i 能够取 $\pm d$ 和 $\pm 3d$,所以最佳幅度检测器要将相关器输出 r 与 4 种可能传输的幅度电平进行比较,并选择在欧氏距离上最接近于 r 的幅度电平。因此,最佳幅度检测器计算距离

$$D_i = |r - A_i|, \qquad i = 0,1,2,3 \tag{5.3.10}$$

并选取对应于最小距离的幅度。

我们注意到,当噪声变量 n 在幅度上超过幅度电平之间距离的一半时,也就是说当 $|n| > d$时,就会发生判决错误。然而,当传输的幅度电平是 $\pm 3d$ 或 $\pm -3d$ 时,差错只会发生在一个方向上。因为这 4 种幅度电平是等概率的,所以一个符号差错的平均概率是:

$$\begin{aligned}
P_4 &= \frac{3}{4}P(|r - A_m| > d) \\
&= \frac{3}{2}\int_d^{\infty}\frac{1}{\sqrt{2\pi}\sigma}\mathrm{e}^{-x^2/(2\sigma^2)}\mathrm{d}x \\
&= \frac{3}{2}\int_{d/\sigma}^{\infty}\frac{1}{\sqrt{2\pi}}\mathrm{e}^{-x^2/2}\mathrm{d}x \\
&= \frac{3}{2}Q\left(\sqrt{\frac{d^2}{\sigma^2}}\right) \\
&= \frac{3}{2}Q\left(\sqrt{\frac{2d^2}{N_0}}\right)
\end{aligned} \tag{5.3.11}$$

可以看出,相继幅度电平之间的平方距离是$(2d)^2 \equiv \delta^2$,因此平均差错概率可以表示为:

$$P_4 = \frac{3}{2}Q\left(\sqrt{\frac{\delta^2}{2N_0}}\right) \tag{5.3.12}$$

另外,平均差错概率还可以利用信号能量来表示。因为全部 4 种幅度电平是等概率的,所以每个符号平均传输的信号能量是:

$$\mathscr{E}_{av} = \frac{1}{4}\sum_{k=1}^{4}\int_0^T s_k^2(t)\mathrm{d}t = 5d^2 \tag{5.3.13}$$

其中,$d^2 = \mathscr{E}_{av}/5$,所以

$$P_4 = \frac{3}{2}Q\left(\sqrt{\frac{2\mathscr{E}_{av}}{5N_0}}\right) \tag{5.3.14}$$

因为每个传输符号由两个信息比特组成,所以每个比特传输的平均能量是$\mathscr{E}_{av}/2 \equiv \mathscr{E}_{avb}$。

---

① 原著中式(5.3.12)为:$P_4 = \frac{3}{2}Q\left[\sqrt{\frac{\delta^2}{4N_0}}\right]$——译者注

图 5.21 给出了作为 SNR(定义为 $10\log_{10}(\mathcal{E}_{avb}/N_0)$)的函数的平均差错概率 P_4。

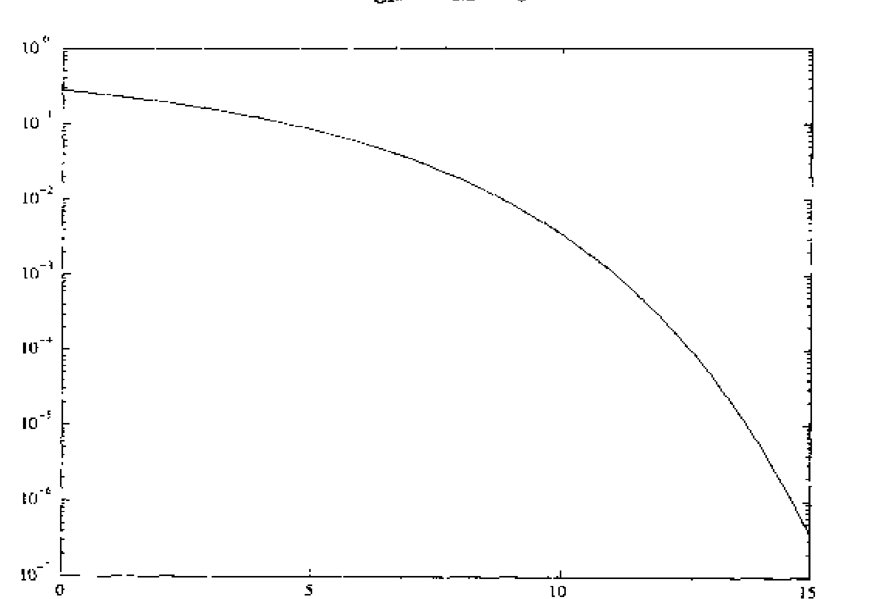


图 5.21　4 电平 PAM 的符号差错概率

解说题

解说题 5.8　[多幅度信号仿真]

对 4 电平 PAM 通信系统进行 Monte Carlo 仿真,该系统使用了一个信号相关器(如前所述),再紧跟着一个幅度检测器。待仿真的系统的模型如图 5.22 所示。

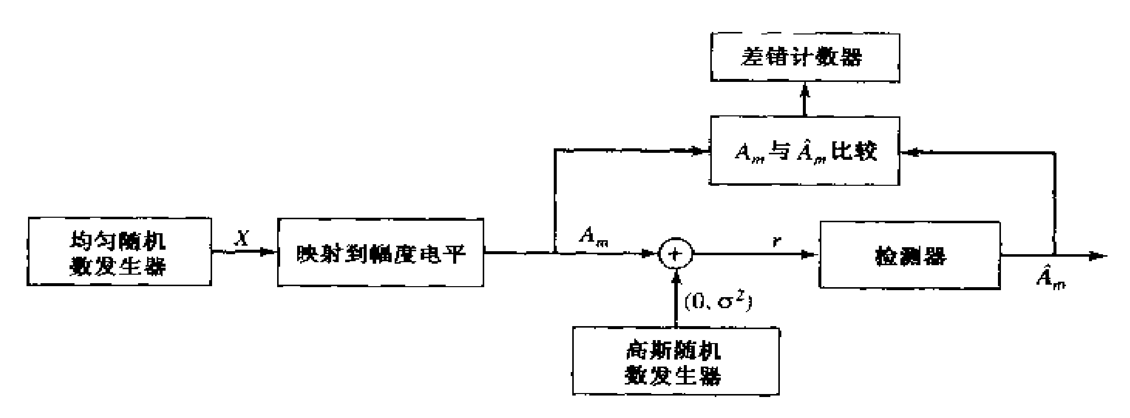


图 5.22　用于 Monte Carlo 仿真的 4 电平 PAM 的方框图

题　解

我们要仿真出随机变量 r 的产生,它是信号相关器的输出和检测器的输入。先要产生一个 4 组符号的序列,再将该序列映射到对应的幅度电平 $\{A_m\}$。为了完成这个任务,用一个随机数发生器产生在范围(0,1)内的均匀随机数。然后,将这个范围再分成 4 个相等的区间(0,0.25),(0.25,0.5),(0.5,0.75)和(0.75,1.0)。这些子区间分别对应于 4 个符号(信息比特对)00,01,11 和 10。这样,均匀随机数发生器的输出就分别映射到相应的信号幅度电平($-3d$, $-d$, d, $3d$)。

高斯随机数发生器产生零均值、方差 σ^2 的加性噪声分量。为了方便,可以将距离参数归一化到 $d=1$ 而使 σ^2 变化。检测器观察到 $r=A_m+n$,并计算 r 和 4 种可能的传输信号幅度之间的距离。它的输出 $\hat{A}_m$ 相应于最小距离的信号幅度电平。将 $\hat{A}_m$ 与真正的传输信号幅度进行比较,差错计数器用来对检测器产生的差错计数。

图 5.23 是在不同的平均比特 SNR 值下,传输 $N=10\ 000$ 个符号时的仿真结果,这个平均比特 SNR 定义为:

$$\frac{\mathcal{E}_{\text{avb}}}{N_0}=\frac{5}{4}\left(\frac{d^2}{\sigma^2}\right) \tag{5.3.15}$$

由该图可注意到仿真结果与由式(5.3.14)计算出的 p_4 理论值的一致性。

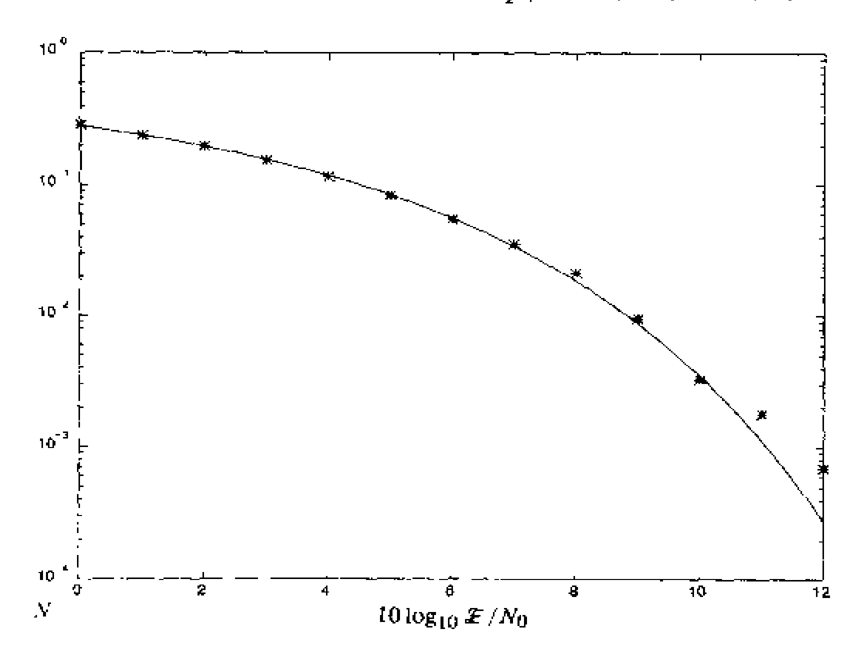


图 5.23　Monte Carlo 仿真的差错概率与 $M=4$ 时 PAM 的理论差错概率的比较

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrated Problem 5.8.

echo on
SNRindB1=0:1:12;
SNRindB2=0:0.1:12;
for i=1:length(SNRindB1),
  % simulated error rate
  smld_err_prb(i)=smldPe58(SNRindB1(i));
  echo off;
end;
echo on;
for i=1:length(SNRindB2),
  % signal-to-noise ratio
  SNR_per_bit=exp(SNRindB2(i)*log(10)/10);
  % theoretical error rate
  theo_err_prb(i)=(3/2)*Qfunct(sqrt((4/5)*SNR_per_bit));
  echo off;
```

```
end;
echo on;
% Plotting commands follow.
semilogy(SNRindB1,smld_err_prb,'*');
hold
semilogy(SNRindB2,theo_err_prb);
```

m 文件

```
function [p]=smldPe58(snr_in_dB)
% [p]=smldPe58(snr_in_dB)
%               SMLDPE58  simulates the probability of error for the given
%               snr_in_dB, signal to noise ratio in dB.
d=1;
SNR=exp(snr_in_dB*log(10)/10);          % signal to noise ratio per bit
sgma=sqrt((5*d^2)/(4*SNR));             % sigma, standard deviation of noise
N=10000;                                % number of symbols being simulated
% Generation of the quaternary data source follows.
for i=1:N,
  temp=rand;                            % a uniform random variable over (0,1)
  if (temp<0.25),
    dsource(i)=0;                       % With probability 1/4, source output is "00"
  elseif (temp<0.5),
    dsource(i)=1;                       % With probability 1/4, source output is "01."
  elseif (temp<0.75),
    dsource(i)=2;                       % With probability 1/4, source output is "10."
  else
    dsource(i)=3;                       % With probability 1/4, source output is "11."
  end
end;
% detection, and probability of error calculation
numoferr=0;
for i=1:N,
  % the matched filter outputs
  if (dsource(i)==0),
    r=-3*d+gngauss(sgma);               % if the source output is "00"
  elseif (dsource(i)==1),
    r=-d+gngauss(sgma);                 % if the source output is "01"
  elseif (dsource(i)==2)
    r=d+gngauss(sgma);                  % if the source output is "10"
  else
    r=3*d+gngauss(sgma);                % if the source output is "11"
  end;
  % Detector follows.
  if (r<-2*d),
    decis=0;                            % Decision is "00"
  elseif (r<0),
    decis=1;                            % Decision is "01."
  elseif (r<2*d),
    decis=2;                            % Decision is "10."
  else
    decis=3;                            % Decision is "11."
  end;
  if (decis~=dsource(i)),               % If it is an error, increase the error counter.
    numoferr=numoferr+1;
```

```
  end;
end;
p=numoferr/N;                          % probability of error estimate
```

5.3.3 多幅度电平的信号波形

构造多于4电平的多幅度信号是相当直接的。一般来说，一组 $M=2^k$ 的多幅度信号波形表示为：

$$s_m(t)=A_m g(t),\qquad 0\leqslant t\leqslant T,\qquad m=0,1,2,\cdots,M-1$$

其中，M 个幅度值是等间隔的，给出为：

$$A_m=(2m-M+1)d,\quad m=0,1,\cdots,M-1 \tag{5.3.16}$$

$g(t)$ 是一个矩形脉冲，由式(5.3.2)定义。每个信号波形携带有 $k=\log_2 M$ 信息比特。当比特率是 $R=1/T_b$ 时，相应的符号率就是 $\frac{1}{T}=1(kT_b)$。和4电平 PAM 的情况相同，最佳接收机由信号相关器(或匹配滤波器)紧跟一个幅度检测器组成，该幅度检测器计算由式(5.3.10)给出的 $m=0,1,\cdots,M-1$ 时的欧氏距离。对于等概率的幅度电平来说，判决是朝向对应于最小距离的幅度电平做出的。很容易证明，M 电平 PAM 系统最佳检测器的差错概率是：

$$P_M=\frac{2(M-1)}{M}Q\left(\sqrt{\frac{6(\log_2 M)\mathcal{E}_{avb}}{(M^2-1)N_0}}\right) \tag{5.3.17}$$

其中 $\mathcal{E}_{avb}$ 是1个信息比特的平均能量。图5.24给出了 $M=2,4,8,16$ 时的一个符号差错的概率。

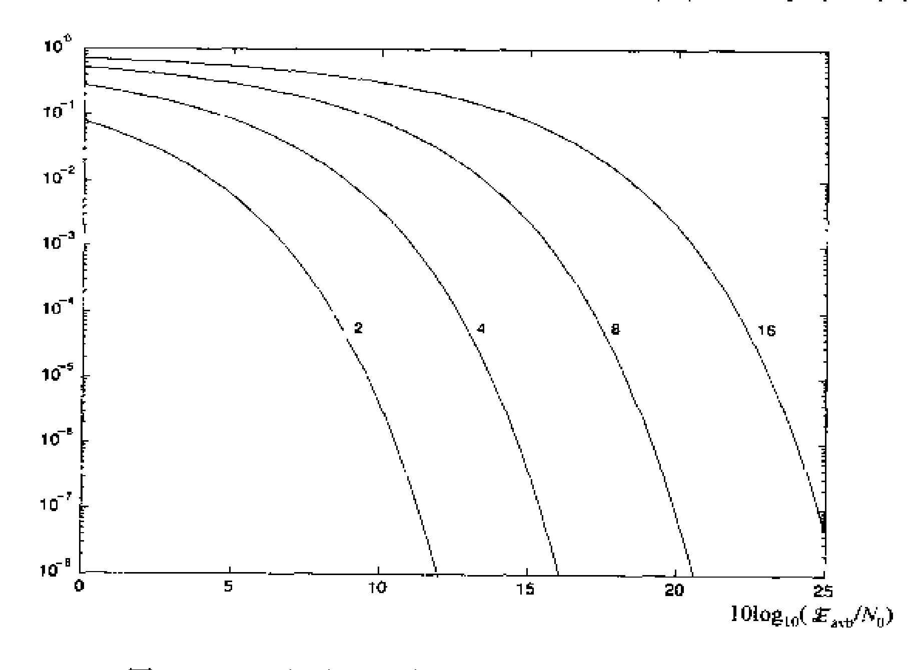


图5.24 M 电平 PAM 在 $M=2,4,8,16$ 时的符号差错概率

解说题

解说题 5.9 [PAM 仿真]

完成16电平 PAM 数字通信系统的 Monte Carlo 仿真，并测量它的误码率性能。

题　解

一般可使用图 5.22 所示的基本方框图。用均匀随机数发生器产生信息符号序列，该序列被当做 4 个信息比特一组。将区间(0,1)划分为 16 个等宽度的子区间可直接产生这 16 组符号，并将这 16 组符号映射到 16 个信号幅度。将一个高斯白噪声序列加到这 16 组信息符号序列上，以形成信号加噪声馈给检测器。检测器按式(5.3.10)给出的关系计算距离，并选择出对应于最小距离的幅度。检测器的输出与真正传输的信息符号序列进行比较，然后将差错计数。图 5.25 给出了当传输 10 000 个符号时所测出的符号误码率，并与 $M=16$ 时由式(5.3.17)给出的理论上的符号误码率进行比较。

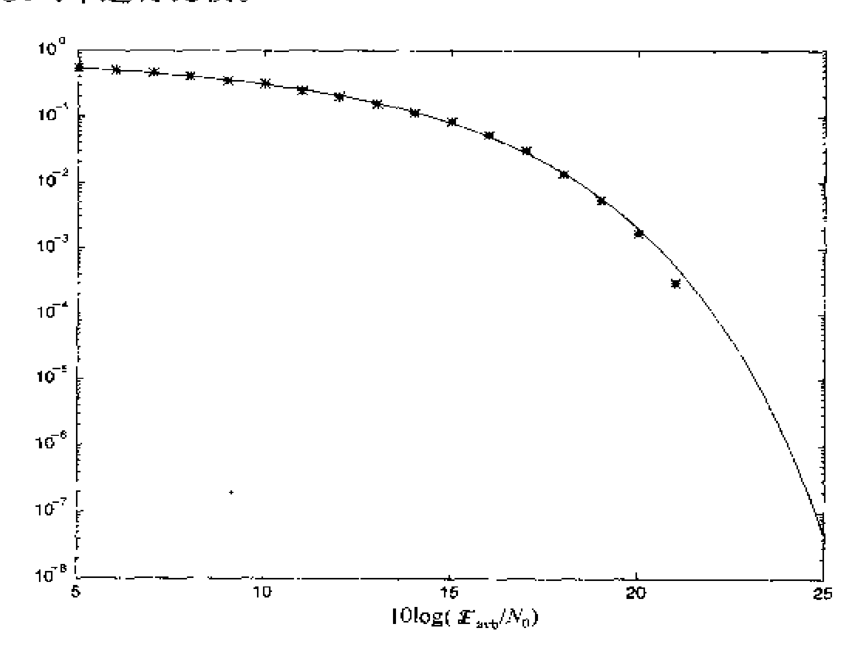


图 5.25　当 $M=16$ PAM 时，Monte Carlo 仿真误码率与理论差错概率的比较

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 5.9.
echo on
SNRindB1=5:1:25;
SNRindB2=5:0.1:25;
M=16;
for i=1:length(SNRindB1),
  % simulated error rate
  smld_err_prb(i)=smldPe59(SNRindB1(i));
  echo off;
end;
echo on ;
for i=1:length(SNRindB2),
  SNR_per_bit=exp(SNRindB2(i)*log(10)/10);
  % theoretical error rate
  theo_err_prb(i)=(2*(M-1)/M)*Qfunct(sqrt((6*log2(M)/(M^2-1))*SNR_per_bit));
  echo off;
end;
echo on;
```

```
% Plotting commands follow.
semilogy(SNRindB1,smld_err_prb,'*');
hold
semilogy(SNRindB2,theo_err_prb);
```

m 文件

```
function [p]=smldPe59(snr_in_dB)
% [p]=smldPe59(snr_in_dB)
%               SMLDPE59  simulates the error probability for the given
%               snr_in_dB, signal-to-noise ratio in dB.
M=16;                                   % 16-ary PAM
d=1;
SNR=exp(snr_in_dB*log(10)/10);          % signal-to-noise ratio per bit
sgma=sqrt((85*d^2)/(8*SNR));            % sigma, standard deviation of noise
N=10000;                                % number of symbols being simulated
% generation of the data source
for i=1:N,
   temp=rand;                           % a uniform random variable over (0,1)
   index=floor(M*temp);                 % The index is an integer from 0 to M-1, where
                                        % all the possible values are equally likely.
   dsource(i)=index;
end;
% detection, and probability of error calculation
numoferr=0;
for i=1:N,
  % matched filter outputs
  % (2*dsource(i)-M+1)*d is the mapping to the 16-ary constellation.
  r=(2*dsource(i)-M+1)*d+gngauss(sgma);
  % the detector
  if (r>(M-2)*d),
    decis=15;
  elseif (r>(M-4)*d),
    decis=14;
  elseif (r>(M-6)*d),
    decis=13;
  elseif (r>(M-8)*d),
    decis=12;
  elseif (r>(M-10)*d),
    decis=11;
  elseif (r>(M-12)*d),
    decis=10;
  elseif (r>(M-14)*d),
    decis=9;
  elseif (r>(M-16)*d),
    decis=8;
  elseif (r>(M-18)*d),
    decis=7;
  elseif (r>(M-20)*d),
    decis=6;
  elseif (r>(M-22)*d),
    decis=5;
  elseif (r>(M-24)*d),
    decis=4;
  elseif (r>(M-26)*d),
    decis=3;
  elseif (r>(M-28)*d),
    decis=2;
```

```
  elseif (r>(M-30)*d),
    decis=1;
  else
    decis=0;
  end;
  if (decis~=dsource(i)),    % If it is an error, increase the error counter.
    numoferr=numoferr+1;
  end;
end;
p=numoferr/N;                % probability of error estimate
```

5.4　多维信号

在前一节中,我们构造了多幅度信号波形,因此可以在每个信号波形上传输多比特信息由此,具有 $M=2^k$ 幅度电平的信号波形就能够在每个信号波形上传输 $k=\log_2 M$ 信息比特。另外,我们还看到多幅度信号在几何上能表示成实线上的一些信号点(见图 5.20)。这样的信号波形称为**一维信号**。

这一节要考虑一类 $M=2^k$ 的信号波形,它们具有一个多维表示;也就是说,这组信号波形在几何上能用 N 维空间的点来表示。前面已经讨论过,二进制正交信号在几何上能用二维空间上的点来表示。

5.4.1　多维正交信号

有很多方法能够构造多维信号波形,它们具有各自的性质。这一节考虑的一组 $M=2^k$ 信号波形 $s_i(t), i=0,1,\cdots,M-1$ 具有如下性质:(a) 互为正交;(b) 等能量。这两个性质可以简洁地表示为:

$$\int_0^T s_i(t)s_k(t)\mathrm{d}t=\mathscr{E}\delta_{ik},\qquad i,k=0,1,\cdots,M-1 \tag{5.4.1}$$

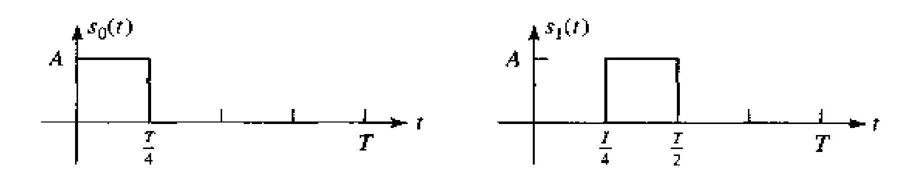
其中$\mathscr{E}$是每个信号波形的能量,δ_{ik}是单位脉冲函数,定义为:

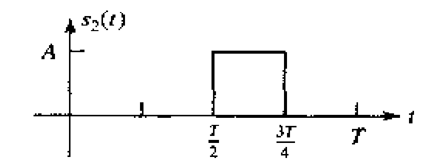
$$\delta_{ik}=\begin{cases}1, & i=k\\ 0, & i\neq k\end{cases} \tag{5.4.2}$$

和前面讨论的一样,假设某个信息源正提供一个信息比特序列,要将这个序列经由某个通信信道传输。该信息比特以 R b/s 的均匀码率出现,R 的倒数就是比特区间 T_b。调制器每次取 k 比特,并将它映射到 $M=2^k$ 个信号波形中的一个。每个 k 比特组称为 1 个符号。传输 1 个符号可用的时间区间是 $T=kT_b$,所以 T 是符号区间。

在区间$(0,T)$内构造一组 $M=2^k$ 的等能量正交波形的最简单方法是将这个区间分成 M 个持续期为 T/M 的相等子区间,并对每个子区间指定一种信号波形。图 5.26 就是对于 $M=4$ 的这样一种构造的信号。以这种方式构造的所有信号波形都具有相同的能量:

$$\begin{aligned}\mathscr{E}&=\int_0^T s_i^2(t)\mathrm{d}t,\qquad i=0,1,2,\cdots,M-1\\ &=\frac{A^2T}{M}\end{aligned} \tag{5.4.3}$$





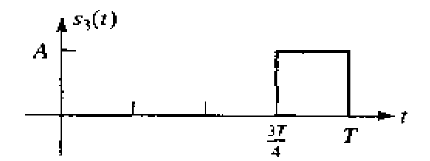


图 5.26 4 个正交、等能量信号波形的例子

这样一组正交波形可以表示成一组 M 维的正交向量,即

$$
\begin{aligned}
s_0 &= (\sqrt{\mathcal{E}},0,0,\cdots,0) \\
s_1 &= (0,\sqrt{\mathcal{E}},0,\cdots,0) \\
&\vdots \\
s_M &= (0,0,\cdots,0,\sqrt{\mathcal{E}})
\end{aligned}
\tag{5.4.4}
$$

图 5.27 给出了相应于 $M=2$ 和 $M=3$ 的正交信号的信号星座图。

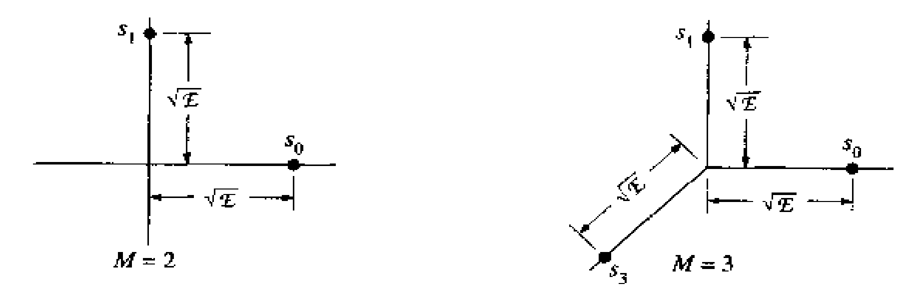


图 5.27 $M=2$ 和 $M=3$ 的正交信号的信号星座图

现在假定用这些正交信号波形经由 AWGN 信道传输信息。这样,如果传输的是 $s_i(t)$,则接收到的信号波形就是

$$r(t)=s_i(t)+n(t), \qquad 0\leqslant t\leqslant T, \qquad i=0,1,\cdots,M-1 \tag{5.4.5}$$

这里 $n(t)$是功率谱为 $N_0/2$(W/Hz)的高斯白噪声过程的一个样本函数。接收机观察到信号 $r(t)$并做出判决:M 个信号波形中的哪一个被传输。

AWGN 信道的最佳接收机

使差错概率最小的接收机首先将信号 $r(t)$通过一组并联的 M 个匹配滤波器或 M 个相关器。因为信号相关器和匹配滤波器在采样瞬时产生相同的输出,所以只考虑用信号相关器的情况,如图 5.28 所示。

信号相关器

接收信号 $r(t)$与 M 个信号波形中的每个做互相关,相关器输出在 $t=T$ 采样,因此 M 个

相关器的输出是：

$$r_i = \int_0^T r(t)s_i(t)\mathrm{d}t, \qquad i = 0,1,\cdots,M-1 \tag{5.4.6}$$

它可以用向量表示成 $r=[r_0,r_1,\cdots,r_{M-1}]^t$。假设传输的是信号波形 $s_0(t)$，那么：

$$r_0 = \int_0^T s_0^2(t)\mathrm{d}t + \int_0^T n(t)s_0(t)\mathrm{d}t = \mathscr{E} + n_0 \tag{5.4.7}$$

且

$$\begin{aligned} r_i &= \int_0^T s_0(t)s_i(t)\mathrm{d}t + \int_0^T n(t)s_i(t)\mathrm{d}t \\ &= \int_0^T n(t)s_i(t)\mathrm{d}t = n_i, \qquad i = 1,2,3,\cdots,M-1 \end{aligned} \tag{5.4.8}$$

其中，

$$n_i = \int_0^T n(t)s_i(t)\mathrm{d}t \tag{5.4.9}$$

因此，输出 r_0 由一个信号分量$\mathscr{E}$和一个噪声分量 n_0 组成，而其余的 $M-1$ 个输出仅由噪声组成。这个噪声分量是高斯型的，均值为零，方差为：

$$\begin{aligned} \sigma^2 &= E(n_i^2) \\ &= \int_0^T\int_0^T s_i(t)s_i(\tau)E[n(t)n(\tau)]\mathrm{d}t\mathrm{d}\tau \\ &= \frac{N_0}{2}\int_0^T\int_0^T s_i(t)s_i(\tau)\delta(t-\tau)\mathrm{d}t\mathrm{d}\tau \\ &= \frac{N_0}{2}\int_0^T s_i^2(t)\mathrm{d}t \\ &= \frac{N_0\mathscr{E}}{2} \end{aligned} \tag{5.4.10}$$

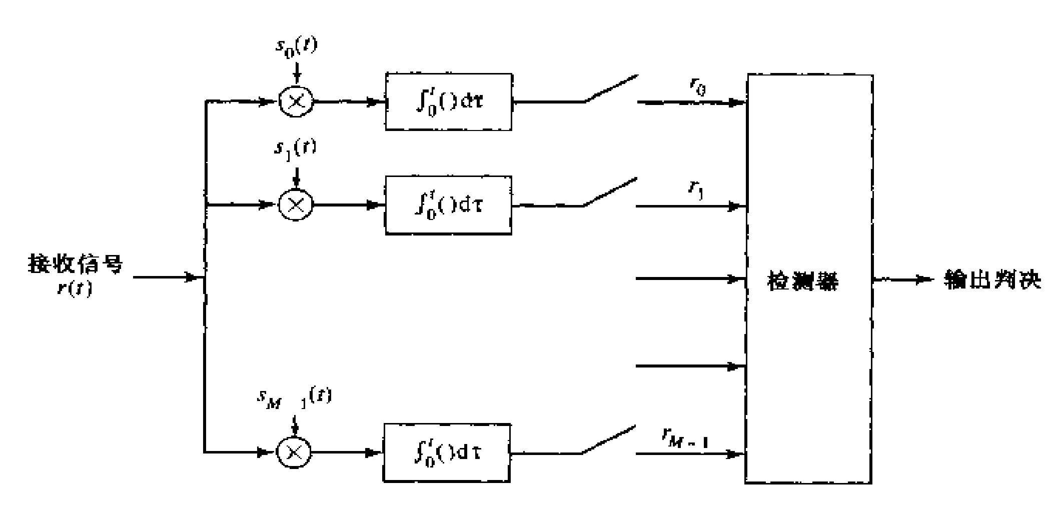


图 5.28 多维正交信号的最佳接收机

读者可以证明 $E(n_in_j)=0, i\neq j$。结果，这些相关器输出的概率密度函数是：

$$p(r_0 \mid s_0(t)\text{被传输}) = \frac{1}{\sqrt{2\pi}\sigma}\mathrm{e}^{-(r_0-\mathscr{E})^2/(2\sigma^2)}$$

$$p(r_i \mid s_0(t)\text{被传输}) = \frac{1}{\sqrt{2\pi}\sigma} e^{-r_i^2/(2\sigma^2)}, \qquad i = 1,2,\cdots,M-1$$

检测器

最佳检测器观察到 M 个相关器的输出 r_i, $i = 0,1,\cdots,M-1$,并选出对应于最大相关器输出的信号。在 $s_0(t)$ 被传输的情况下,正确判决的概率就是 $r_0 > r_i$ 的概率,$i = 1,2,\cdots,M-1$。或者

$$P_c = P(r_0 > r_1, r_0 > r_2, \cdots, r_0 > r_{M-1}) \tag{5.4.11}$$

1 个符号差错的概率就是:

$$P_M = 1 - P_c = 1 - P(r_0 > r_1, r_0 > r_2, \cdots, r_0 > r_{M-1}) \tag{5.4.12}$$

可以证明,P_M 能表示成如下积分形式:

$$P_M = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \{1 - [1 - Q(y)]^{M-1}\} e^{-(y-\sqrt{2\mathscr{E}/N_0})^2/2} dy \tag{5.4.13}$$

对于 $M = 2$ 的特殊情况,式(5.4.13)简化为:

$$P_2 = Q\left(\sqrt{\frac{\mathscr{E}_b}{N_0}}\right)$$

这就是在 5.2 节对二进制正交信号所得的结果。

当传输其他 $M-1$ 个信号中的任何一个时,可以得到差错概率的同一表达式。因为全部 M 个信号都是等概率的,所以由式(5.4.13)给出的 P_M 表达式就是 1 个符号差错的平均概率。这个积分式可用数值法求出。

有时我们希望把符号差错概率转换成一个二进制数字差错的等效概率。对于等概率的正交信号,所有符号差错都是等概率的,并且其发生概率为:

$$\frac{P_M}{M-1} = \frac{P_M}{2^k - 1} \tag{5.4.14}$$

而且,在差错中有 $\binom{k}{n}$(k 中的 n 个比特)种可能方式,所以每 k 比特符号的平均比特差错数是:

$$\sum_{n=1}^{k} n \binom{k}{n} \frac{P_M}{2^{k-1}} = k \frac{2^{k-1}}{2^k - 1} P_M \tag{5.4.15}$$

而平均比特差错概率就是式(5.4.15)的结果除以 k,即每符号的比特数,因此,

$$P_b = \frac{2^{k-1}}{2^k - 1} P_M \tag{5.4.16}$$

图 5.29 给出了 $M = 2,4,8,16,32,64$ 时的 1 个二进制数字差错的概率作为每比特 SNR ($\mathscr{E}_b/N_0$)的函数的图,这里 $\mathscr{E}_b = \mathscr{E}/k$ 是每比特的能量。这个图说明,增加波形的数目 M 可以降低为达到某个给定的比特差错概率所需的每比特 SNR。

计算由式(5.4.13)给出的差错概率的 MATLAB 脚本如下所示。

---

① 原著中此处为 $\frac{1}{\sqrt{2\pi}} e^{-r_i^2/(2\sigma^2)}$ ——译者注

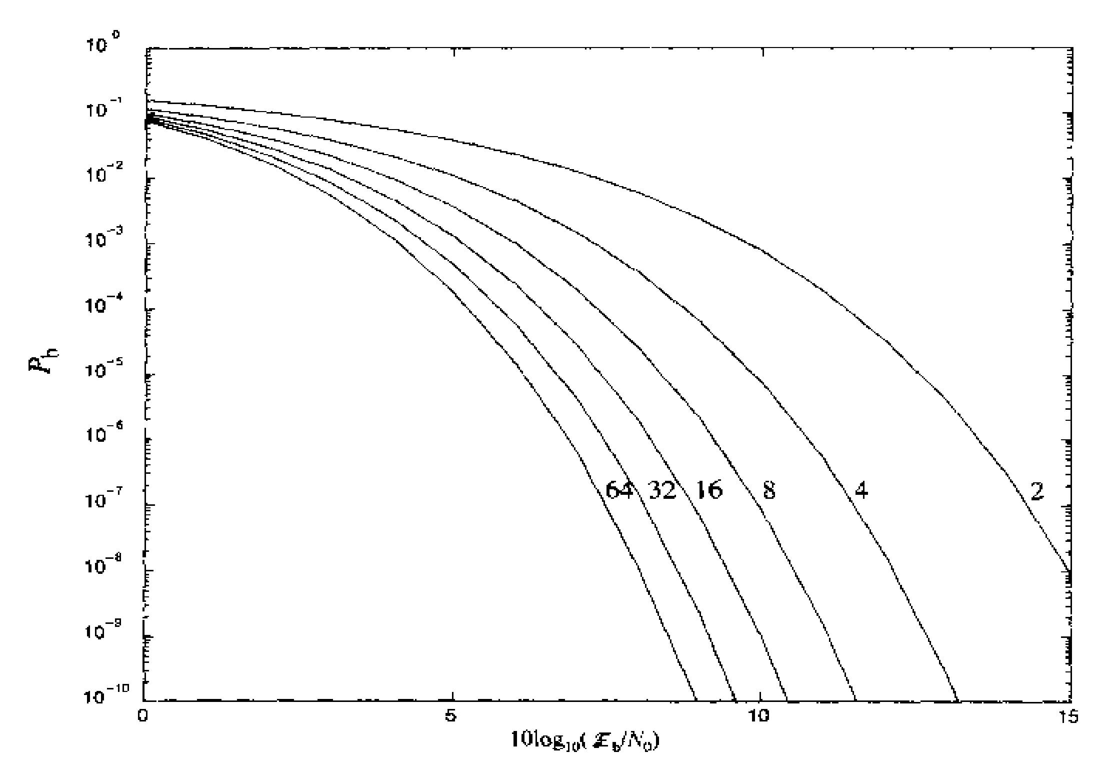


图 5.29　正交信号的比特差错概率

m 文件

```
% MATLAB script that generates the probability of error versus the signal-to-noise ratio
initial_snr=0;
final_snr=15;
snr_step=1;
tolerance=1e-7;                        % tolerance used for the integration
minus_inf=-20;                         % This is practically negative infinity.
plus_inf=20;                           % This is practically infinity.
snr_in_dB=initial_snr:snr_step:final_snr;
for i=1:length(snr_in_dB),
    snr=10^(snr_in_dB(i)/10);
    Pe_2(i)=Qfunct(sqrt(snr));
    Pe_4(i)=(2/3)*quad8('bdt_int',minus_inf,plus_inf,tolerance,[],snr,4);
    Pe_8(i)=(4/7)*quad8('bdt_int',minus_inf,plus_inf,tolerance,[],snr,8);
    Pe_16(i)=(8/15)*quad8('bdt_int',minus_inf,plus_inf,tolerance,[],snr,16);
    Pe_32(i)=(16/31)*quad8('bdt_int',minus_inf,plus_inf,tolerance,[],snr,32);
    Pe_64(i)=(32/63)*quad8('bdt_int',minus_inf,plus_inf,tolerance,[],snr,64);
end;
% Plotting commands follow.
```

解说题

解说题 5.10　[正交信号仿真]

完成 $M=4$ 的正交信号的数字通信系统 Monte Carlo 仿真，待仿真系统的模型如图 5.30 所示。

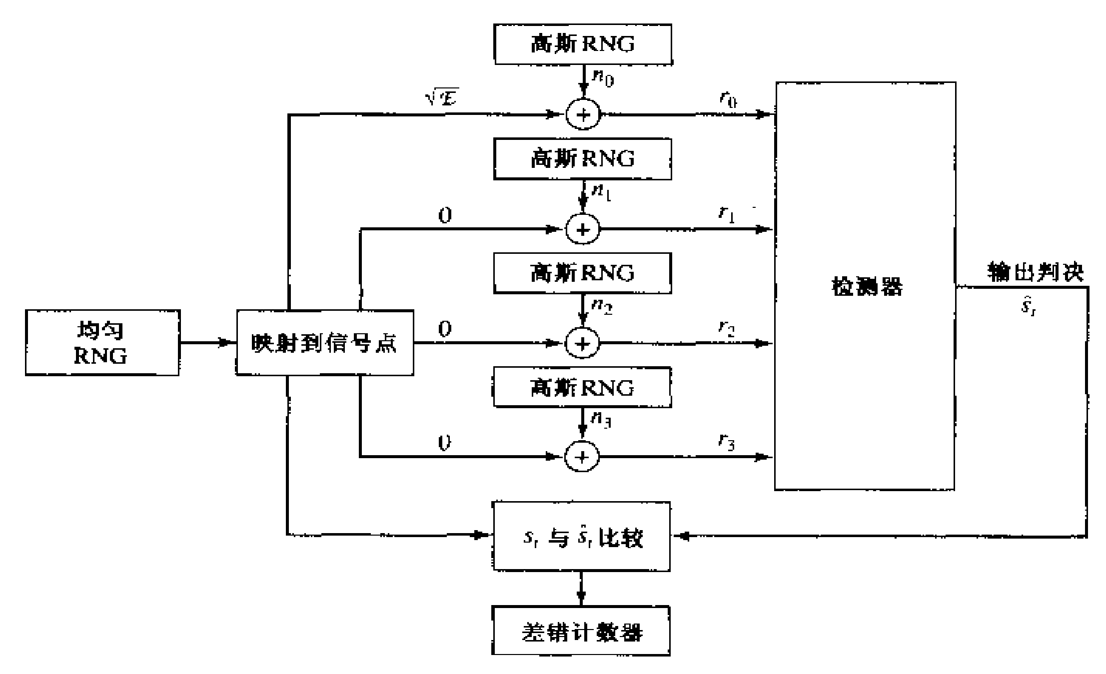


图 5.30 用于 Monte Carlo 仿真的 $M=4$ 的正交信号系统的方框图

题 解

先要仿真随机变量 r_0, r_1, r_2 和 r_3,它们构成了检测器的输入。首先产生一个 0 和 1 的二进制序列,正如解说题 5.4 所述,它们是等概率出现并互为统计独立的。将这个二进制序列组成比特对,这些比特对再映射到对应的信号分量。产生单个比特对的另一种办法是像解说题 5.8那样产生比特对。在任意一种情况下,都将这 4 个符号映射为如下的信号点:

$$
\begin{aligned}
00 &\to s_0 = (\sqrt{\mathscr{E}},0,0,0)\\
01 &\to s_1 = (0,\sqrt{\mathscr{E}},0,0)\\
10 &\to s_2 = (0,0,\sqrt{\mathscr{E}},0)\\
11 &\to s_3 = (0,0,0,\sqrt{\mathscr{E}})
\end{aligned}
\tag{5.4.17}
$$

加性噪声分量 n_0, n_1, n_2 和 n_3 由 4 个高斯噪声发生器产生,每个都有零均值和方差 $\sigma^2 = N_0\mathscr{E}/2$。为了方便,可将符号能量归一化到$\mathscr{E}=1$,而使 σ^2 变化。因为$\mathscr{E}=2\mathscr{E}_b$ 所以有$\mathscr{E}_b=\frac{1}{2}$。检测器的输出与传输的比特序列进行比较,差错计数器用于对比特差错数计数。

图 5.31 给出了在几个不同的 SNR($\mathscr{E}_b/N_0$)值下,传输 20 000 个比特时的仿真结果。注意仿真结果与由式(5.4.16)给出的理论 P_b 值的一致性。

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 5.10.
echo on
SNRindB=0:2:10;
for i=1:length(SNRindB),
  % simulated error rate
  smld_err_prb(i)=smldP510(SNRindB(i));
```

```
  echo off;
end;
echo on;
% Plotting commands follow.
semilogy(SNRindB,smld_err_prb,'*');
```

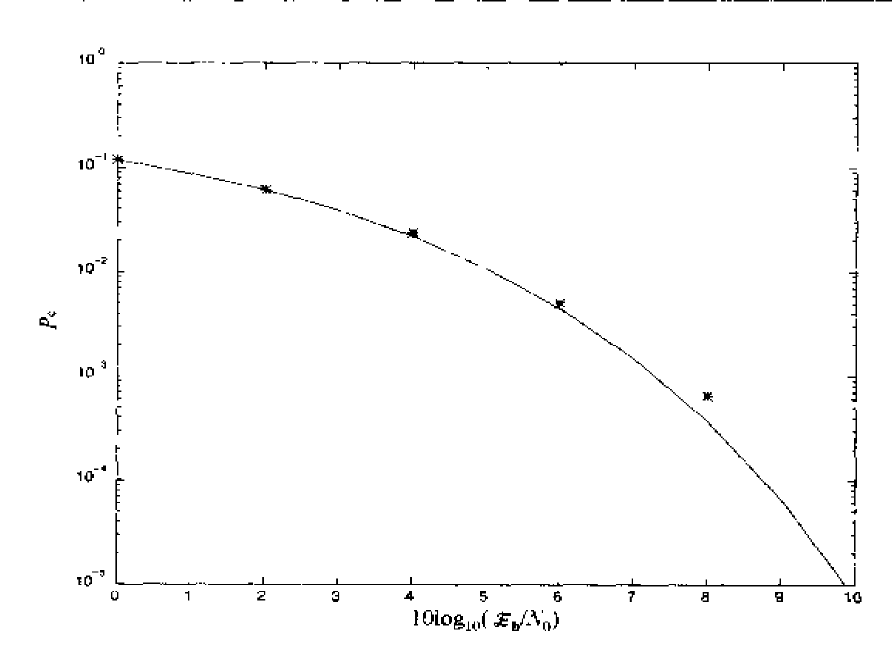


图 5.31　对 $M=4$ 的正交信号，由 Monte Carlo 仿真的比特差错概率与理论差错概率的比较

m 文件

```
function [p]=smldP510(snr_in_dB)
% [p]=smldP510(snr_in_dB)
%               SMLDP510  simulates the probability of error for the given
%               snr_in_dB, signal-to-noise ratio in dB.
M=4;                                  % quaternary orthogonal signaling
E=1;
SNR=exp(snr_in_dB*log(10)/10);        % signal-to-noise ratio per bit
sgma=sqrt(E^2/(4*SNR));               % sigma, standard deviation of noise
N=10000;                              % number of symbols being simulated
% generation of the quaternary data source
for i=1:N,
  temp=rand;                          % a uniform random variable over (0,1)
  if (temp<0.25),
    dsource1(i)=0;
    dsource2(i)=0;
  elseif (temp<0.5),
    dsource1(i)=0;
    dsource2(i)=1;
  elseif (temp<0.75),
    dsource1(i)=1;
    dsource2(i)=0;
  else
    dsource1(i)=1;
    dsource2(i)=1;
  end
  end;
  % detection, and probability of error calculation
```

```
numoferr=0;
for i=1:N,
  % matched filter outputs
  if ((dsource1(i)==0) & (dsource2(i)==0)),
    r0=sqrt(E)+gngauss(sgma);
    r1=gngauss(sgma);
    r2=gngauss(sgma);
    r3=gngauss(sgma);
  elseif ((dsource1(i)==0) & (dsource2(i)==1)),
    r0=gngauss(sgma);
    r1=sqrt(E)+gngauss(sgma);
    r2=gngauss(sgma);
    r3=gngauss(sgma);
  elseif ((dsource1(i)==1) & (dsource2(i)==0)),
    r0=gngauss(sgma);
    r1=gngauss(sgma);
    r2=sqrt(E)+gngauss(sgma);
    r3=gngauss(sgma);
  else
    r0=gngauss(sgma);
    r1=gngauss(sgma);
    r2=gngauss(sgma);
    r3=sqrt(E)+gngauss(sgma);
  end;
  % the detector
  max_r=max([r0 r1 r2 r3]);
  if (r0==max_r),
    decis1=0;
    decis2=0;
  elseif (r1==max_r),
    decis1=0;
    decis2=1;
  elseif (r2==max_r),
    decis1=1;
    decis2=0;
  else
    decis1=1;
    decis2=1;
  end;
  % Count the number of bit errors made in this decision.
  if (decis1~=dsource1(i)),              % If it is an error, increase the error counter.
    numoferr=numoferr+1;
  end;
  if (decis2~=dsource2(i)),              % If it is an error, increase the error counter.
    numoferr=numoferr+1;
  end;
end;
p=numoferr/(2*N);                        % bit error probability estimate
```

5.4.2 双正交信号

正如在前一节中所看到的,一组 $M=2^k$ 个等能量的正交波形可以通过将符号区间 T 划分成区间为 T/M 的 M 个相等的子区间,并对每个子区间指定一个矩形信号,来构成另一组 $M=2^k$ 个多维信号,它们具有双正交(biorthogonal)性质。在这样的信号集合中,一半波形是正交的、而另一半则是这些正交波形的负值;也就是说,$s_0(t),s_1(t),\cdots,s_{(M/2)-1}(t)$都是正交波形,

而其他 $M/2$ 波形就是 $s_{i+M/2}(t)=-s_i(t)$，$i=0,1,\cdots,(M/2)-1$。由此，得到 M 个信号，每个具有 $M/2$ 维。

这 $M/2$ 个正交波形可以很容易地通过将符号区间 $T=kT_b$ 划分成 $M/2$ 个不重叠的子区间，每个子区间的持续期为 $2T/M$，再将每个子区间指定一个矩形脉冲来实现。图 5.32 说明了用这种方式构造的 $M=4$ 的一组双正交波形。以这种方式构造的 M 个信号的几何表示可用如下 $M/2$ 维的信号点给出：

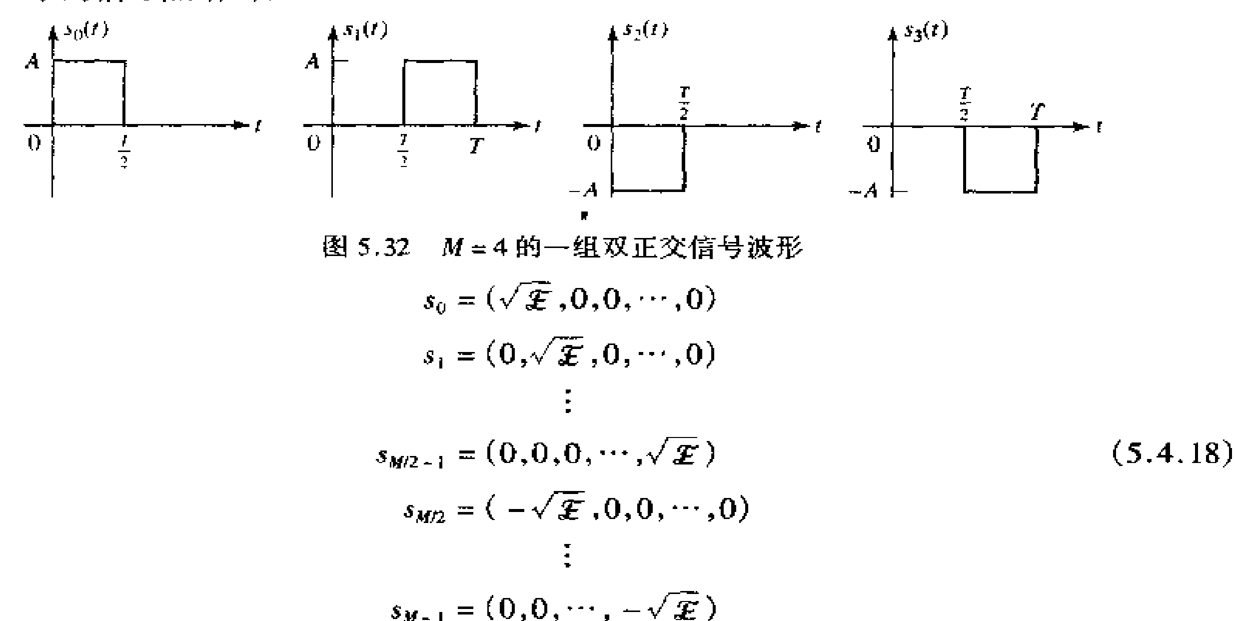


图 5.32　$M=4$ 的一组双正交信号波形

$$
\begin{aligned}
s_0&=(\sqrt{\mathcal{E}},0,0,\cdots,0)\\
s_1&=(0,\sqrt{\mathcal{E}},0,\cdots,0)\\
&\vdots\\
s_{M/2-1}&=(0,0,0,\cdots,\sqrt{\mathcal{E}})\\
s_{M/2}&=(-\sqrt{\mathcal{E}},0,0,\cdots,0)\\
&\vdots\\
s_{M-1}&=(0,0,\cdots,-\sqrt{\mathcal{E}})
\end{aligned}
\tag{5.4.18}
$$

与正交信号的情况一样，假设是经由 AWGN 信道用双正交信号传输信息，那么接收到的信号波形可以表示成：

$$r(t)=s_i(t)+n(t),\qquad 0\leqslant t\leqslant T \tag{5.4.19}$$

这里 $s_i(t)$是传输的波形，$n(t)$是功率谱为 $N_0/2$(W/Hz)的高斯白噪声过程的一个样本函数。

最佳接收机

通过将接收信号 $r(t)$与 $M/2$ 个双正交信号波形中的每个做互相关，在 $t=T$ 对相关器输出采样，并将 $M/2$ 个相关器输出通过检测器，即可实现最佳接收机。据此有：

$$r_i=\int_0^T r(t)s_0(t)\mathrm{d}t\qquad i=0,1,\cdots,\frac{M}{2}-1 \tag{5.4.20}$$

假定传输的信号波形是 $s_0(t)$，那么

$$
\begin{aligned}
r_i&=\int_0^T r(t)s_0(t)\mathrm{d}t\qquad i=0,1,\cdots,\frac{M}{2}-1\\
&=\begin{cases}\mathcal{E}+n_0, & i=0\\ n_i, & i\neq 0\end{cases}
\end{aligned}
\tag{5.4.21}
$$

其中，

$$n_i=\int_0^T n(t)s_i(t)\mathrm{d}t\qquad i=0,1,\cdots,\frac{M}{2}-1 \tag{5.4.22}$$

$\mathcal{E}$是每个信号波形的符号能量。噪声分量是零均值高斯噪声，方差为 $\sigma^2=\mathcal{E}N_0/2$。

检测器

检测器观察到 $M/2$ 个相关器的输出 $\{r_i, 0 \leqslant i \leqslant (M/2)-1\}$，并选出 $|r_i|$ 最大的相关器输出。假设

$$|r_j| = \max_i \{|r_i|\} \tag{5.4.23}$$

那么，若 $r_j > 0$，则检测器就选出信号 $s_j(t)$，若 $r_j < 0$，则选出 $-s_j(t)$。

为了确定差错概率，假设传输的是 $s_0(t)$，那么一个正确判决概率就等于 $r_0 = \mathscr{E} + n_0 > 0$ 和 $|r_0| > |r_i|, i = 1,2,\cdots, M/2-1$ 的概率。因此

$$P_c = \int_0^\infty \left[\frac{1}{\sqrt{2\pi}} \int_{-r_0\sqrt{\mathscr{E}N_0/2}}^{r_0\sqrt{\mathscr{E}N_0/2}} e^{-x^2/2} dx \right]^{M-1} p(r_0) dr_0 \tag{5.4.24}$$

其中

$$p(r_0) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(r_0-\mathscr{E})^2/(2\sigma^2)} \tag{5.4.25}$$

最后，1 个符号差错的概率就是：

$$P_M = 1 - P_c \tag{5.4.26}$$

P_c 和 P_M 可由式(5.4.24)和式(5.4.25)对不同的 M 个值用数值法求出。图 5.33 给出了 $M = 2,4,8,16$ 和 32 时 P_M 作为信噪比 $\mathscr{E}_b/N_0$ 的函数的图，从中可以看出这个图与正交信号的图是相似的。然而，对于双正交信号来说，要注意 $P_4 > P_2$。这是由于在图 5.33 中画出的是符号差错概率 P_M，如果画出等效比特差错概率，就会发现 $M = 2$ 和 $M = 4$ 是重合的。

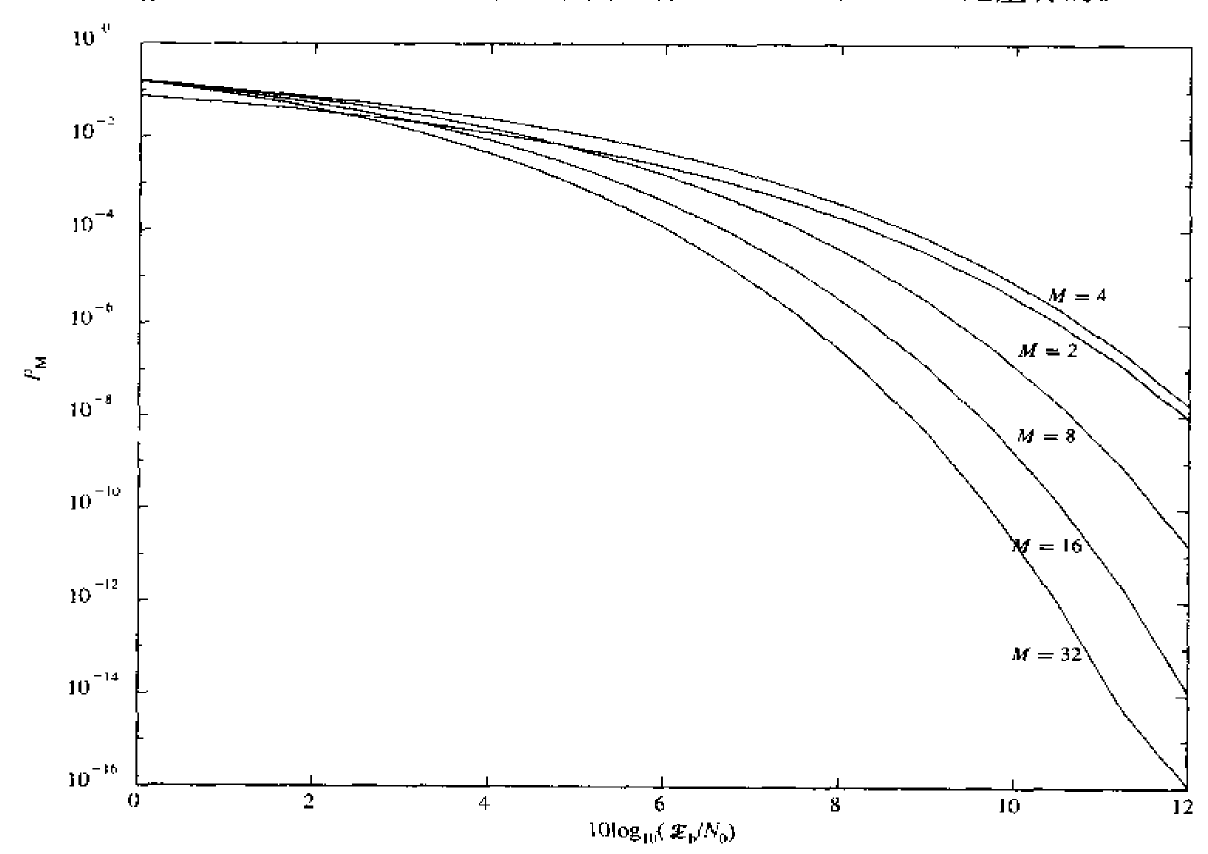


图 5.33 双正交信号的符号差错概率

根据式(5.4.24)和式(5.4.25)计算差错概率的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script that generates the probability of error versus the signal-to-noise ratio.
initial_snr=0;
final_snr=12;
snr_step=0.75;
tolerance=eps;                          % tolerance used for the integration
plus_inf=20;                            % This is practically infinity.
snr_in_dB=initial_snr:snr_step:final_snr;
for i=1:length(snr_in_dB),
    snr=10^(snr_in_dB(i)/10);
    Pe_2(i)=1-quad8('bdt_int2',0,plus_inf,tolerance,[],snr,2);
    Pe_4(i)=1-quad8('bdt_int2',0,plus_inf,tolerance,[],snr,4);
    Pe_8(i)=1-quad8('bdt_int2',0,plus_inf,tolerance,[],snr,8);
    Pe_16(i)=1-quad8('bdt_int2',0,plus_inf,tolerance,[],snr,16);
    Pe_32(i)=1-quad8('bdt_int2',0,plus_inf,tolerance,[],snr,32);
end;
% Plotting commands follow.
```

解说题

解说题 5.11　[双正交信号仿真]

完成 $M=4$ 的双正交信号数字通信系统的 Monte Carlo 仿真，待仿真的系统的模型如图 5.34所示。

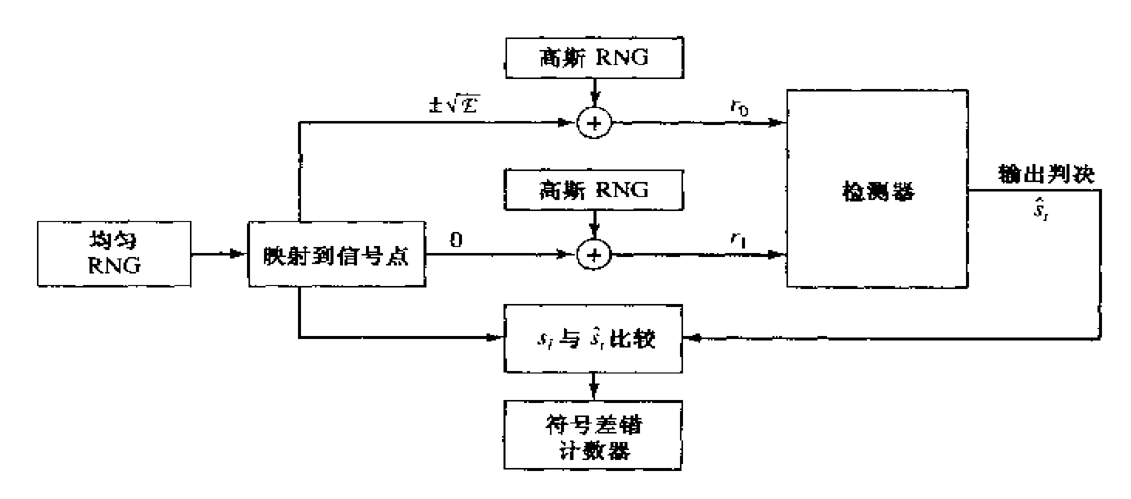


图 5.34　用于 Monte Carlo 仿真的 $M=4$ 的双正交信号系统的方框图

题　解

如图 5.34 所示，先要仿真随机变量 r_0 和 r_1，它们构成了检测器的输入。正如解说题 5.4，首先产生一个 0 和 1 的二进制序列，它们是等概率的并互为统计独立。这个二进制序列按成对比特组合，然后按下面的关系映射为相应的信号分量：

$$00 \to s_0=(\sqrt{\mathcal{E}},0)$$

$$01 \to s_1=(0,\sqrt{\mathcal{E}})$$

$$10 \rightarrow s_2 = (0, -\sqrt{\mathcal{E}})$$

$$11 \rightarrow s_3 = (-\sqrt{\mathcal{E}}, 0)$$

另外也可以用解说题 5.8 中的方法直接产生 2 比特的符号。

因为 $s_2 = -s_1$ 和 $s_3 = -s_0$，所以解调仅需要两个相关器或匹配滤波器，它们的输出是 r_0 和 r_1。由两个高斯噪声发生器产生加性噪声分量 n_0 和 n_1，每个都有零均值和方差 $\sigma^2 = N_0\mathcal{E}/2$。为了方便，可以将符号能量归一化到$\mathcal{E} = 1$ 而使 σ^2 改变。因为$\mathcal{E} = 2\mathcal{E}_b$，所以有$\mathcal{E}_b = \frac{1}{2}$。检测器输出与传输的比特序列进行比较，用差错计数器对符号差错数和比特差错计数。

图 5.35 给出了在几个不同的 SNR($\mathcal{E}_b/N_0$)值时，传输 20 000 个比特的仿真结果。应该注意到，仿真结果和由式(5.4.26)和式(5.4.24)给出 P_4 理论值的一致性。

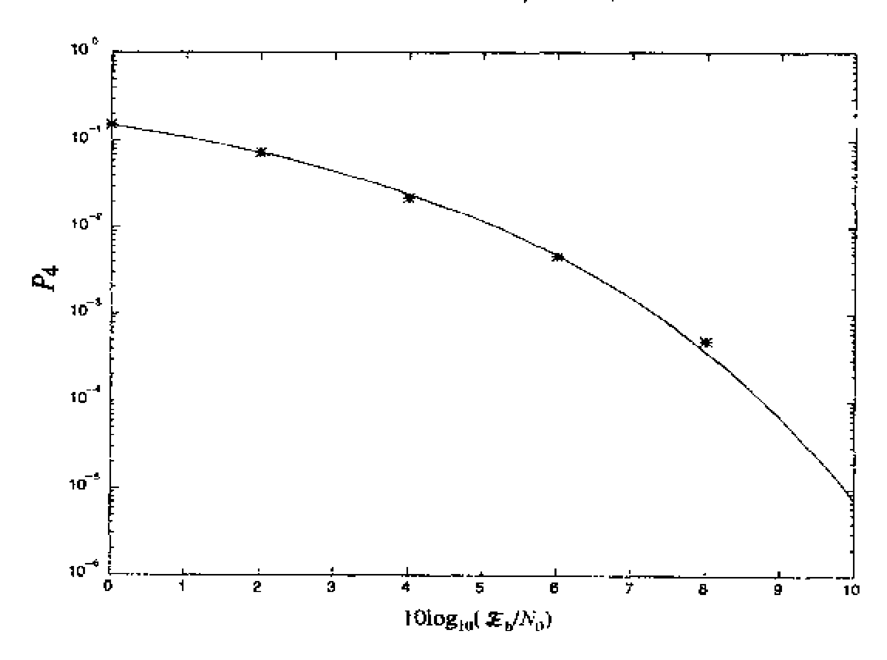


图 5.35 Monte Carlo 仿真对 $M = 4$ 的双正交信号的符号差错概率与理论差错概率的比较

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 5.11.
echo on
SNRindB=0:2:10;
for i=1:length(SNRindB),
  % simulated error rate
  smld_err_prb(i)=smldP511(SNRindB(i));
  echo off;
end;
echo on ;
% Plotting commands follow.
```

m 文件

```
function [p]=smldP511(snr_in_dB)
% [p]=smldP511(snr_in_dB)
%                SMLDP511  simulates the probability of error for the given
%                snr_in_dB, signal-to-noise ratio in dB, for the system
%                described in Illustrative Problem 5.11.
M=4;                                        % quaternary biorthogonal signaling
E=1;
SNR=exp(snr_in_dB*log(10)/10);              % signal-to-noise ratio per bit
sgma=sqrt(E^2/(4*SNR));                     % sigma, standard deviation of noise
N=10000;                                    % number of symbols being simulated
% generation of the quaternary data source
for i=1:N,
   temp=rand;                               % uniform random variable over (0,1)
   if (temp<0.25),
      dsource(i)=0;
   elseif (temp<0.5),
      dsource(i)=1;
   elseif (temp<0.75),
      dsource(i)=2;
   else
      dsource(i)=3;
   end
end;
% detection, and error probability computation
numoferr=0;
for i=1:N,
   % the matched filter outputs
   if (dsource(i)==0)
      r0=sqrt(E)+gngauss(sgma);
      r1=gngauss(sgma);
   elseif (dsource(i)==1)
      r0=gngauss(sgma);
      r1=sqrt(E)+gngauss(sgma);
   elseif (dsource(i)==2)
      r0=-sqrt(E)+gngauss(sgma);
      r1=gngauss(sgma);
   else
      r0=gngauss(sgma);
      r1=-sqrt(E)+gngauss(sgma);
   end;
   % detector follows
   if (r0>abs(r1)),
      decis=0;
   elseif (r1>abs(r0)),
      decis=1;
   elseif (r0<-abs(r1)),
      decis=2;
   else
      decis=3;
   end;
   if (decis~=dsource(i)),                  % If it is an error, increase the error counter.
      numoferr=numoferr+1;
   end;
end;
p=numoferr/N;                               % bit error probability estimate
```

5.5 习题

5.1 假定用图5.2所示的两个正交信号经由一个AWGN信道传输二进制信息,在持续期 T_b 的每个比特区间接收到的信号由式(5.2.1)给出。假设接收信号波形以 $10/T_b$ 速率采样,即每比特区间内10个样本,所以在离散时间内幅度为 A 的信号波形 $s_0(t)$ 用10个样本 $(A,A,\cdots,A)$ 表示,而信号波形 $s_1(t)$ 由10个样本 $(A,A,A,A,A,-A,-A,-A,-A,-A)$ 表示,这样,当传输 $s_0(t)$ 时,接收序列的采样形式是:

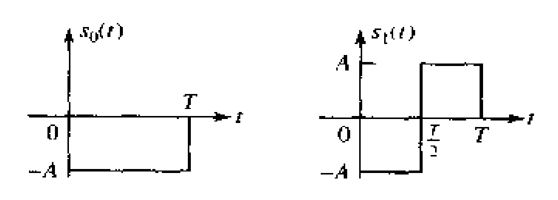
$$r_k = A + n_k, \qquad k = 1,2,\cdots,10$$

而当传输 $s_1(t)$ 时,接收序列的采样形式是:

$$r_k = \begin{cases} A + n_k, & 1 \leqslant k \leqslant 5 \\ -A + n_k & 6 \leqslant k \leqslant 10 \end{cases}$$

其中,序列 $\{n_k\}$ 是独立同分布的零均值,每个随机变量具有方差为 σ^2 的高斯过程。写一个MATLAB程序,它对两种可能接收信号的每一种产生序列 $\{r_k\}$,在不同的加性高斯噪声方差值 $\sigma^2=0,\sigma^2=0.1,\sigma^2=1.0$ 和 $\sigma^2=2.0$ 下,完成序列 $\{r_k\}$ 与由采样形式表示的两种可能信号 $s_0(t)$ 和 $s_1(t)$ 中的每一种的离散时间相关。信号幅度可以归一化到 $A=1$。在时刻 $k=1,2,3,\cdots,10$,画出相关器的输出。

5.2 对于图P5.2所示的两个信号波形 $s_0(t)$ 和 $s_1(t)$,重做习题5.1。描述这两个信号和图P5.2中的两个信号之间的相似和差异之处。从传输二进制信息信号的序列的角度来看,一组比另一组信号更好一些吗?



图P5.2

5.3 这个习题的目的是,用两个匹配滤波器来替换习题5.1中的两个相关器,产生信号的情况与习题5.1相同。

写一个MATLAB程序,它对两种可能接收信号中的每一种产生序列 $\{r_k\}$,在不同的加性高斯噪声方差值 $\sigma^2=0,\sigma^2=0.1,\sigma^2=1.0$ 和 $\sigma^2=2.0$ 的情况下,完成序列 $\{r_k\}$ 与由采样形式表示的两种可能信号 $s_0(t)$ 和 $s_1(t)$ 中的每一种的离散时间匹配滤波,信号幅度可以归一化到1。在对应于 $k=1,2,\cdots,10$ 的时间瞬时,画出相关器的输出。

5.4 对于图P5.2所示的信号波形,重做习题5.3。

5.5 运行一个MATLAB程序,完成图5.8给出的基于正交信号的二进制通信系统Monte Carlo仿真。仿真对10000个比特实行,并测量 $\sigma^2=0,\sigma^2=0.1,\sigma^2=1.0$ 和 $\sigma^2=2.0$ 时的差错概率。画出理论误码率和由Monte Carlo仿真测得的误码率,并比较这两个结果。另外,画出对于每种 σ^2 值在检测器输入端1000个接收到的信号加噪声的样本。

5.6 对图5.13所示的基于反极性信号的二进制通信系统,重做习题5.5。

5.7　对基于反极性信号的二进制通信系统，重做习题 5.5。

5.8　运行一个 MATLAB 程序，完成 $M=4$ 的 PAM 通信系统的仿真。仿真对 10 000 个符号(20 000 比特)实行，并测量在 $\sigma^2=0,\sigma^2=0.1,\sigma^2=1.0$ 和 $\sigma^2=2.0$ 时的符号差错概率。画出理论误码率和由 Monte Carlo 仿真测得的差错，并比较这些结果。另外，画出对于每个 σ^2 值在检测器输入端 1000 个接收到的信号加噪声的样本。

5.9　修改习题 5.8 中的 MATLAB 程序，用于仿真 $M=8$ 的 PAM 的信号，完成 Monte Carlo 仿真的其他数据与习题 5.8 中的相同。

5.10　按解说题 5.10 所述，运行一个 MATLAB 程序，实现 $M=4$ 的正交信号数字通信系统的 Monte Carlo 仿真。完成对 10 000 个符号(20 000 比特)的仿真，并测量在 $\sigma^2=0.1,\sigma^2=1.0$ 和 $\sigma^2=2.0$ 时的比特差错概率。画出理论差错概率和由 Monte Carlo 仿真测得的误码率，并比较这些结果。

5.11　考虑图 P5.11 所示的 4 种信号波形，证明这 4 种信号波形都是互为正交的。习题 5.10 中的 Monte Carlo 仿真结果适用于这些信号吗？为什么？

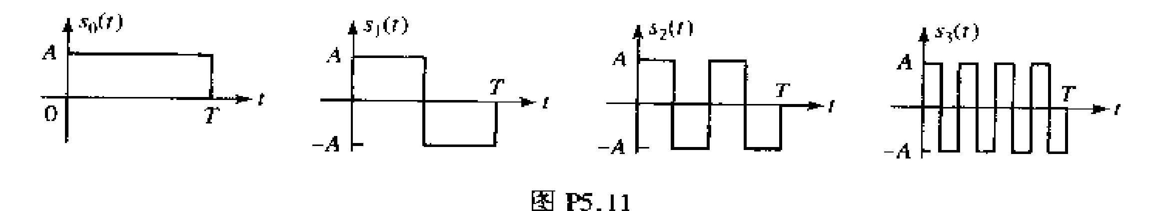


图 P5.11

5.12　按解说题 5.11 所述，运行一个 MATLAB 程序，实现 $M=4$ 的双正交信号数字通信系统的 Monte Carlo 仿真。仿真对 10 000 个符号(20 000 比特)实行，并测量在 $\sigma^2=0.1,\sigma^2=1.0$ 和 $\sigma^2=2.0$ 时的符号差错概率。画出理论符号差错概率和由 Monte Carlo 仿真所测得的误码率，并比较这些结果。另外，画出对于每个 σ^2 值在检测器输入端 1000 个接收到的信号加噪声的样本。

5.13　考虑图 P5.13 所示的 4 种信号波形，证明它们是双正交的。习题 5.12 中的 Monte Carlo 仿真结果适用于这些信号波形吗？为什么？

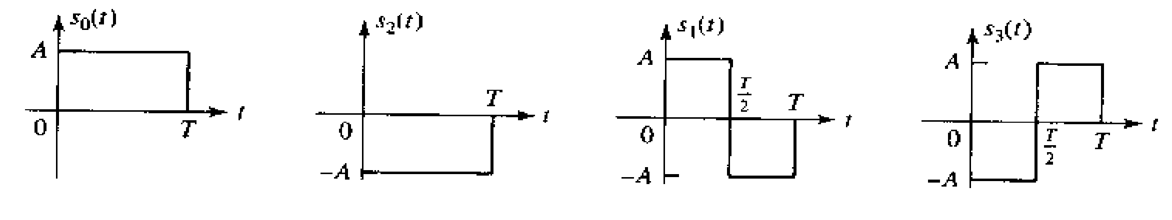


图 P5.13

5.14　利用本章给出的有关图，为了达到一个 10^{-6} 的符号差错概率，试比较对 $M=8$ 的 PAM，正交信号和双正交信号所需的 $\mathcal{E}_b/N_0$ 值。

第 6 章　带限信道的数字传输

6.1　概述

第 6 章讨论的是通过带宽有限信道进行数字传输的几个问题。首先讲述了有关 PAM 信号的频谱特性，接着考虑带限信道的性质和针对这一信道的信号波形设计问题，然后讨论补偿因带限信道引起的失真的信道均衡器设计问题。我们将会指出，信道失真会导致码间干扰(ISI)，这又会引起解调信号的误码。信道均衡器就是一种用于减少码间干扰的装置，由此降低解调数据序列的误码率。

6.2　数字 PAM 信号的功率谱

在前一章中曾考虑过采用脉冲幅度调制(PAM)的数字信息传输，这一节要研究这类信号的频谱特性。

在一条通信信道输入端的数字(PAM)信号一般可表示为：

$$v(t) = \sum_{n=-\infty}^{\infty} a_n g(t - nT) \tag{6.2.1}$$

其中$\{a_n\}$是对应于信源信息符号的幅度序列，$g(t)$是一个脉冲波形，而 T 是符号速率的倒数，T 又称为**符号区间**。$\{a_n\}$中的每个元素是从可能的幅度值中选出来的一个，它们是：

$$A_m = (2m - M + 1)d, m = 0, 1, \cdots, M - 1 \tag{6.2.2}$$

其中，d 是某个加权因子，以决定任意信号幅度对之间的欧氏距离($2d$ 是任意相邻信号幅度电平之间的欧氏距离)。

因为信息序列是一个随机序列，所以对应于信源信息符号的幅度序列$\{a_n\}$也是随机的。这样，PAM 信号 $v(t)$就是随机过程 $V(t)$的一个样本函数。为了确定该随机过程 $V(t)$的频谱特性，就必须求出功率谱。

首先，$V(t)$的均值是：

$$E[V(t)] = \sum_{n=-\infty}^{\infty} E(a_n) g(t - nT) \tag{6.2.3}$$

如果按式(6.2.2)所给出的，选取关于零对称且等概率的信号幅度，那么就有 $E(a_n) = 0$，从而有 $E[V(t)] = 0$。

$V(t)$的自相关函数是：

$$R_v(t + \tau; t) = E[V(t)V(t + \tau)] \tag{6.2.4}$$

有关数字通信的很多教材中都有证明：自相关函数是一个以变量 t 表示的周期为 T 的周期函数。具有周期性均值和周期性自相关函数的随机过程称为周期平稳的随机过程。在单一周期内平均 $R_v(t + \tau; t)$可以将变量 t 消去，即

$$\bar{R}_v(t) = \frac{1}{T}\int_{-T/2}^{T/2} R_v(t+\tau;t)\mathrm{d}t \tag{6.2.5}$$

这个平均自相关函数对 PAM 信号可表示为：

$$\bar{R}_v(\tau) = \frac{1}{T}\sum_{m=-\infty}^{\infty} R_a(m)R_g(\tau - mT) \tag{6.2.6}$$

其中 $R_a(m)=E(a_n a_{n+m})$是序列$\{a_n\}$的自相关，而 $R_g(\tau)$定义为：

$$R_g(\tau) = \int_{-\infty}^{\infty} g(t)g(t+\tau)\mathrm{d}t \tag{6.2.7}$$

$V(t)$的功率谱就是平均自相关函数 $\bar{R}_v(\tau)$的傅里叶变换，即

$$\begin{aligned} S_v(f) &= \int_{-\infty}^{\infty} \bar{R}_v(\tau)\mathrm{e}^{-\mathrm{j}2\pi f\tau}\mathrm{d}t \\ &= \frac{1}{T}S_a(f)\,|G(f)|^2 \end{aligned} \tag{6.2.8}$$

其中$S_a(f)$是幅度序列$\{a_n\}$的功率谱，而 $G(f)$是脉冲 $g(t)$的傅里叶变换。$S_a(f)$定义为：

$$S_a(f) = \sum_{m=-\infty}^{\infty} R_a(m)\mathrm{e}^{-\mathrm{j}2\pi fmT} \tag{6.2.9}$$

由式(6.2.8)可见，PAM 信号的功率谱是信息符号$\{a_n\}$的功率谱和脉冲 $g(t)$的谱的函数。在序列$\{a_n\}$不相关的特殊情况下，即

$$R_a(m) = \begin{cases} \sigma_a^2, m=0 \\ 0, m\neq 0 \end{cases} \tag{6.2.10}$$

其中，$\sigma_a^2 = E(a_n^2)$，就有了对所有的 f，$S_a(f)=\sigma_a^2$，并且

$$S_v(f) = \frac{\sigma_a^2}{T}|G(f)|^2 \tag{6.2.11}$$

这时，$V(t)$的功率谱就完全由脉冲 $g(t)$的频谱特性所决定。

解说题

解说题 6.1　[PAM 功率谱]

当$\{a_n\}$是一个不相关序列，$g(t)$是一个如图 6.1 所示的矩形脉冲时，求 $V(t)$的功率谱。

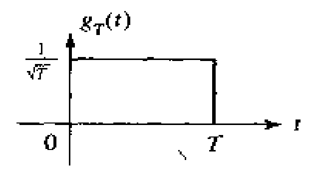


图 6.1　发送脉冲

题　解

$g(t)$的傅里叶变换是：

$$\begin{aligned} G(f) &= \int_{-\infty}^{\infty} g(t)\mathrm{e}^{-\mathrm{j}2\pi ft}\mathrm{d}t \\ &= \sqrt{T}\frac{\sin(\pi fT)}{\pi fT}\mathrm{e}^{-\mathrm{j}\pi fT} \end{aligned} \tag{6.2.12}$$

和

$$S_v(f) = \sigma_a^2 \left(\frac{\sin(\pi f T)}{\pi f T} \right)^2 \tag{6.2.13}$$

这个功率谱如图 6.2 所示。

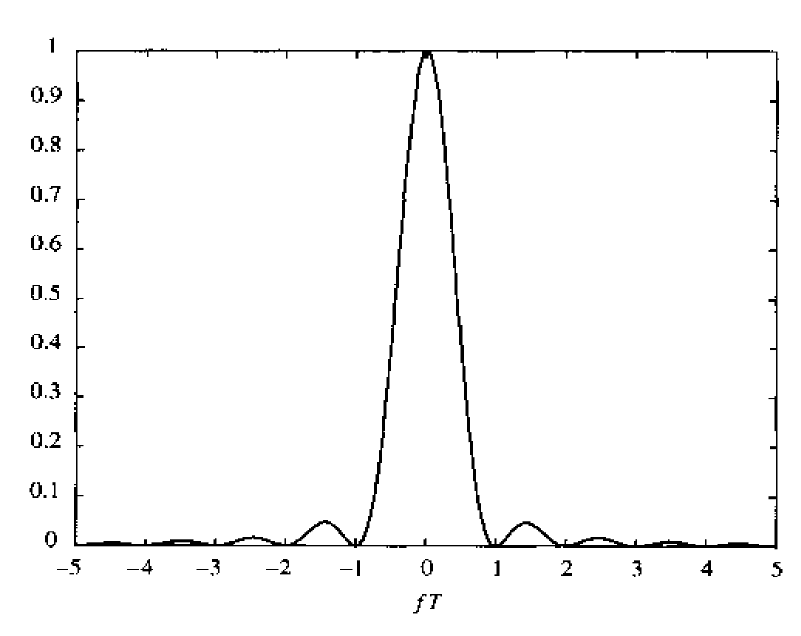


图 6.2 解说题 6.1 中的传输信号的功率谱($\sigma_a^2 = 1$)

计算本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 6.1.
echo on
T=1;
delta_f=1/(100*T);
f=-5/T:delta_f:5/T;
sgma_a=1;
Sv=sgma_a^2*sinc(f*T).^2;
% Plotting command follows.
plot(f,Sv);
```

解说题

解说题 6.2

假设序列$\{a_n\}$的自相关函数是：

$$R_a(m) = \begin{cases} 1, & m = 0 \\ \dfrac{1}{2}, & m = 1, -1 \\ 0, & \text{其余 } m \end{cases} \tag{6.2.14}$$

$g(t)$为如图 6.1 所示的矩形脉冲，求$S_v(f)$。

题 解

PAM 信号 $V(t)$ 的功率谱由式(6.2.8)给出。根据式(6.2.9)和式(6.2.14),序列 $\{a_n\}$ 的功率谱为:

$$\begin{aligned} S_a(f) &= 1 + \cos(2\pi f T) \\ &= 2\cos^2(\pi f T) \end{aligned} \tag{6.2.15}$$

结果得到

$$S_v(f) = 2\cos^2(\pi f T)\left(\frac{\sin(\pi f T)}{\pi f T}\right)^2 \tag{6.2.16}$$

这个功率谱如图 6.3 所示。

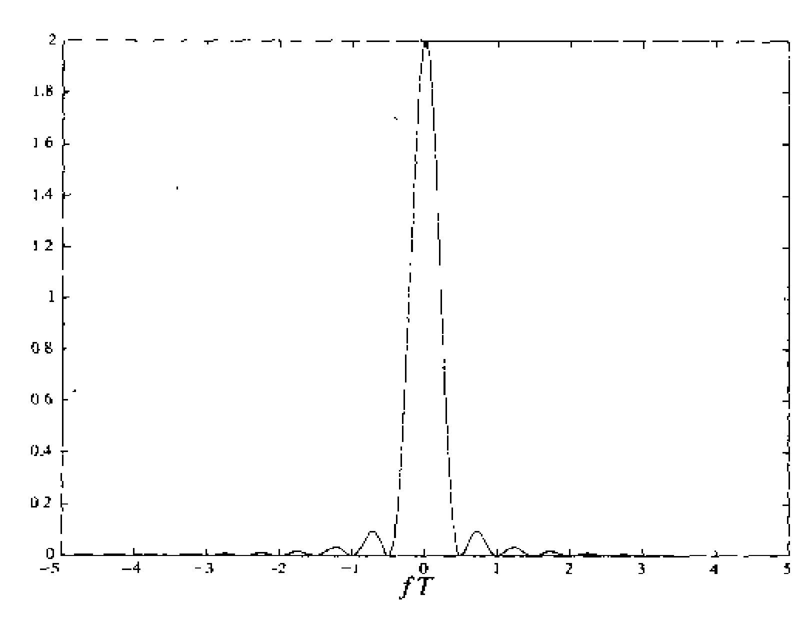


图 6.3 解说题 6.2 中的传输信号的功率谱($\sigma_a^2 = 1$)

完成本题计算的 MATLAB 脚本如下所示。在这种情况下,传输信号 $V(t)$ 的整个功率谱明显地比图 6.2 所示的谱更窄。

m 文件

```
% MATLAB script for Illustrative Problem 6.2.
echo on
T=1;
delta_f=1/(100*T);
f=-5/T:delta_f:5/T;
Sv=2*(cos(pi*f*T).*sinc(f*T)).^2;
% Plotting command follows.
plot(f,Sv);
```

6.3 带限信道特性和信道失真

许多通信信道(其中包括电话信道和某些无线信道)一般都可以用带限线性滤波器来表

征。因此,此类信道可以用它们的频率响应 $C(f)$表示为:

$$C(f) = A(f)e^{j\theta(f)} \tag{6.3.1}$$

其中,$A(f)$称为**幅度响应**,$\theta(f)$称为**相位响应**。有时也用**包络延时**或**群延时**来代替相位特性,其定义为:

$$\tau(f) = -\frac{1}{2\pi}\frac{\mathrm{d}\theta(f)}{\mathrm{d}f} \tag{6.3.2}$$

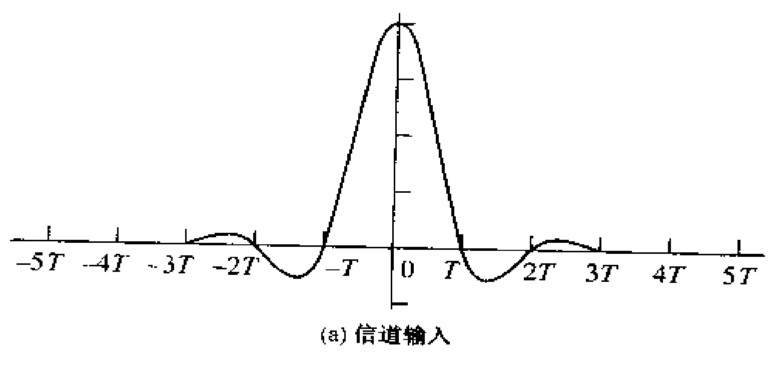
如果在传输信号所占有的带宽 W 内,$A(f)$ = 常数并且 $\theta(f)$是频率的线性函数(或者包络延时 $\tau(f)$ = 常数),就认为信道是**无失真的或理想的**。另一方面,若 $A(f)$和 $\tau(f)$在传输信号占有的带宽内不是常数,那么信道就会使信号失真。若 $A(f)$不是常数,这个失真称为**幅度失真**;若 $\tau(f)$不是常数,则在传输信号上的失真称为**延时失真**。

由非理想信道频率响应特性 $C(f)$引起的幅度和延时失真的一个结果是,在传输信号的速度与信道带宽 W 可比拟的情况下,连续传输的脉冲波形会受到破坏,使得接收端各前后脉冲不能再清晰地分隔开,或者说它们互相重叠了,所以就有了码间干扰。作为延时失真对一个传输脉冲的影响的例子,图 6.4(a)说明了在标定时刻点 $\pm T$、$\pm 2T$ 等处周期性地为零的带限脉冲的情况。当信息是用脉冲幅度携带时(如在 PAM 中),可能传输一个脉冲串,其中每个脉冲都在其他脉冲的周期零值上有一个峰值。然而,经过具有线性包络延时特性 $\theta(f)$的信道传输这个脉冲,$\theta(f)$为二次相位特性,会形成如图 6.4(b)所示的接收脉冲,它的零值交替点不再是周期性分隔的了。结果,持续不断的脉冲序列就会互相破坏,各脉冲的峰值不再清晰可辨。因此,信道延时失真就会形成码间干扰。在本章中将要讨论,有可能在解调器中使用一个滤波器或均衡器来补偿信道的非理想频率响应特性。图 6.4(c)说明了用于补偿信道中的线性失真的一个线性均衡器的输出。

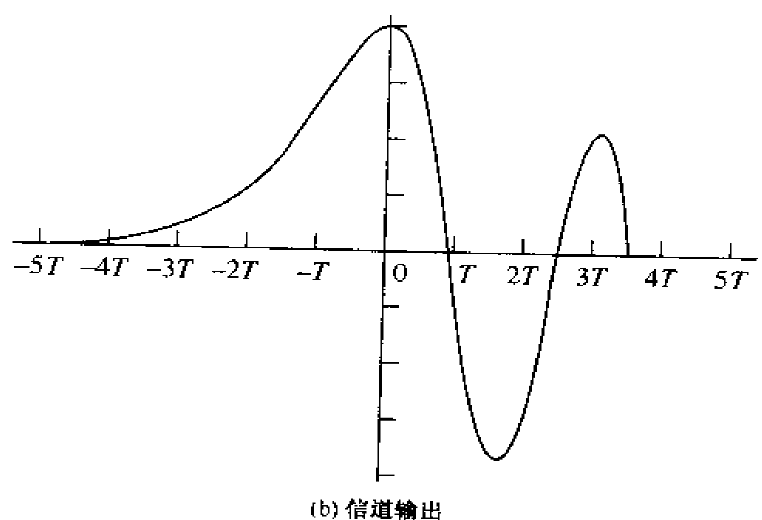
作为一个例子,考虑某个电话信道的码间干扰。图 6.5 显示的是对交换电信网络的一个电话信道测得的平均幅度和延时(作为频率的函数)。我们可以看出,可用的信道频带大约从 300 Hz 到 3200 Hz。这个平均信道的冲激响应如图 6.6 所示,其持续时间大约为 10 ms。通过比较会发现,在这类信道上传输的符号率可以在每秒 2500 个脉冲或符号的量级上,码间干扰可能会达到 20 到 30 个符号以上。

除了电话信道以外,其他一些物理信道也呈现出某些时间弥散的形式,从而引入码间干扰。诸如短波电离层传播(HF)、对流层散射和移动蜂窝无线电等无线信道就是**时间弥散的无线信道**的例子。在这些信道中,时间弥散(从而造成码间干扰)是由于具有不同路径延时的多传播路径引起的。路径的数目和路径之间的相对时间延时都随时间而变化,因此这类无线信道通常称为**时变多径信道**。时变多径的情况会导致各种各样的频率响应特性,所以用于电话信道的频率响应特性对时变多径信道就不合适了。这些无线信道是用散射函数以统计方式表征的;简单地说,散射函数是作为相对时间延时和多普勒频率分布函数的平均接收信号功率的一个二维表示。

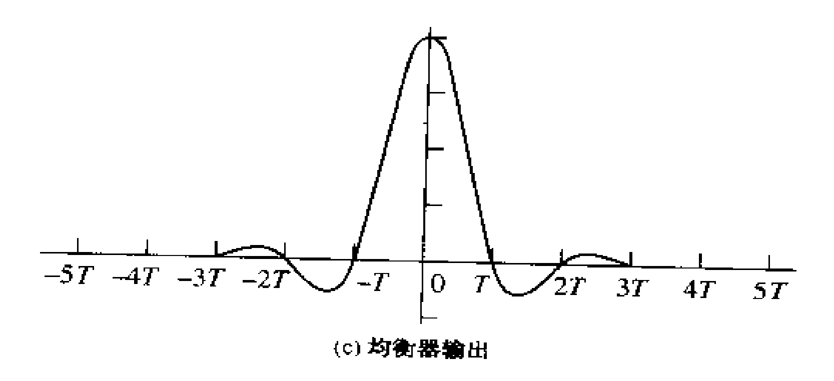
为了进行说明,图 6.7 显示了某个中等距离(150 哩,约为 241.4 km)对流层散射信道测得的某个散射函数。平均信道响应的总持续时间(多径分布)大约为 0.7 μs,在多普勒频率**半功率点**之间的分布在最强的路径上略小于 1 Hz,在其他路径上则稍微大一些。典型的情况是,如果在此类信道上的传输速率是 10^7 符号每秒,而多径扩散在 0.7 μs 将导致跨大约 7 个符号的码间干扰。



(a) 信道输入



(b) 信道输出



(c) 均衡器输出

图 6.4　信道失真的影响

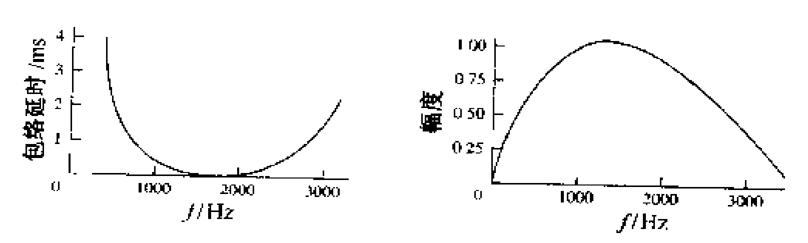


图 6.5　一个中距离电话信道的平均幅度和延时特性

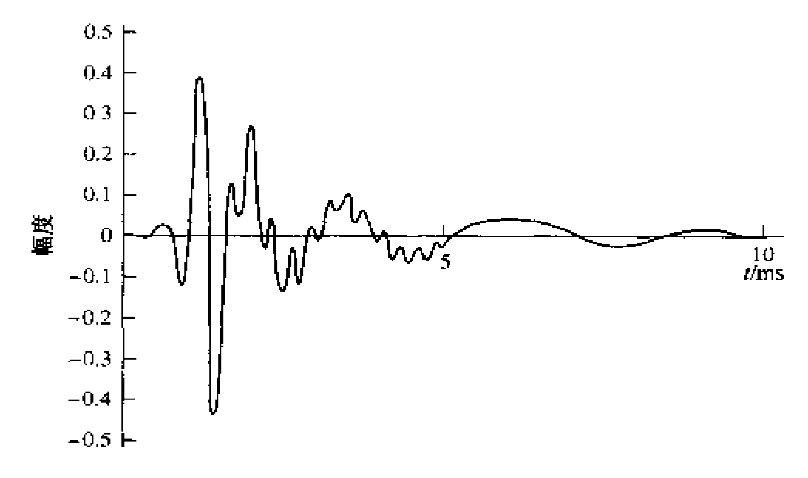


图 6.6　具有图 6.5 所示的幅度和延时特性的平均信道的冲激响应

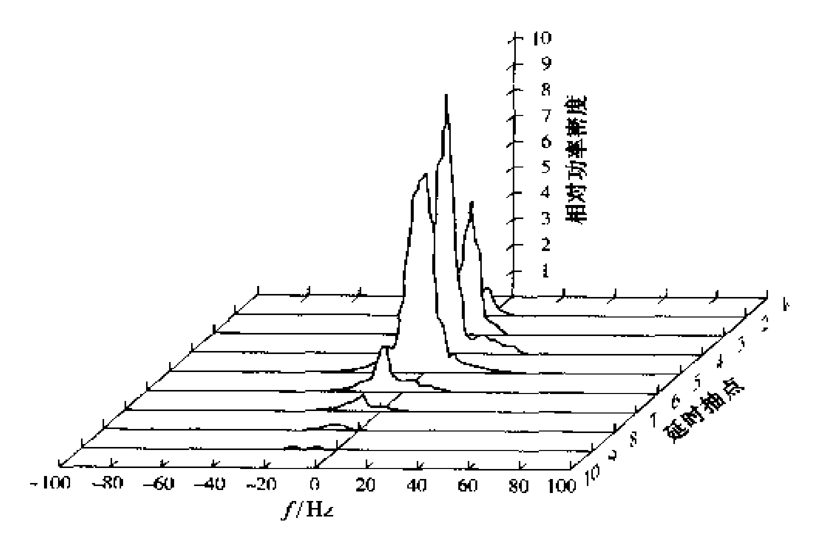


图 6.7　一个中等距离对流层散射信道的散射函数

解说题

解说题 6.3

正如前面指出的，一个带限通信信道可以作为一个线性滤波器来建模，该滤波器的频率响应特性与信道的频率响应特性相匹配。可以用 MATLAB 设计有限脉冲响应(FIR)或无限脉冲响应(IIR)数字滤波器，来近似模拟通信信道的频率响应特性。假定想要对一个具有幅度响应 $A(f)=1, |f|\leqslant 2000$ Hz 和 $A(f)=0, |f|>2000$ Hz，对所有 f 为恒定延时(线性相位)的理想信道建模，数字滤波器的采样频率应选为 $F_s=10\,000$ Hz。因为要求有线性相位，所以只有 FIR 滤波器能满足这个条件。然而，在阻带内不可能达到零响应，因此选择阻带响应为 −40 dB，阻带频率为 2500 Hz。另外，在通带内允许有一个小的量，即 0.5 dB 的起伏。

题　解

满足这些指标要求的，长度 $N=41$ 的 FIR 滤波器的脉冲响应和频率响应如图 6.8 所示。因为 N 是奇数，所以通过滤波器的延时是 $(N+1)/2$ 节抽头，这相应于在采样频率 $F_s=10$ kHz

下$(N+1)/20$ ms 的时间延时。在本例中，FIR 滤波器是采用切比雪夫近似（Remez 算法），用 MATLAB 设计的。

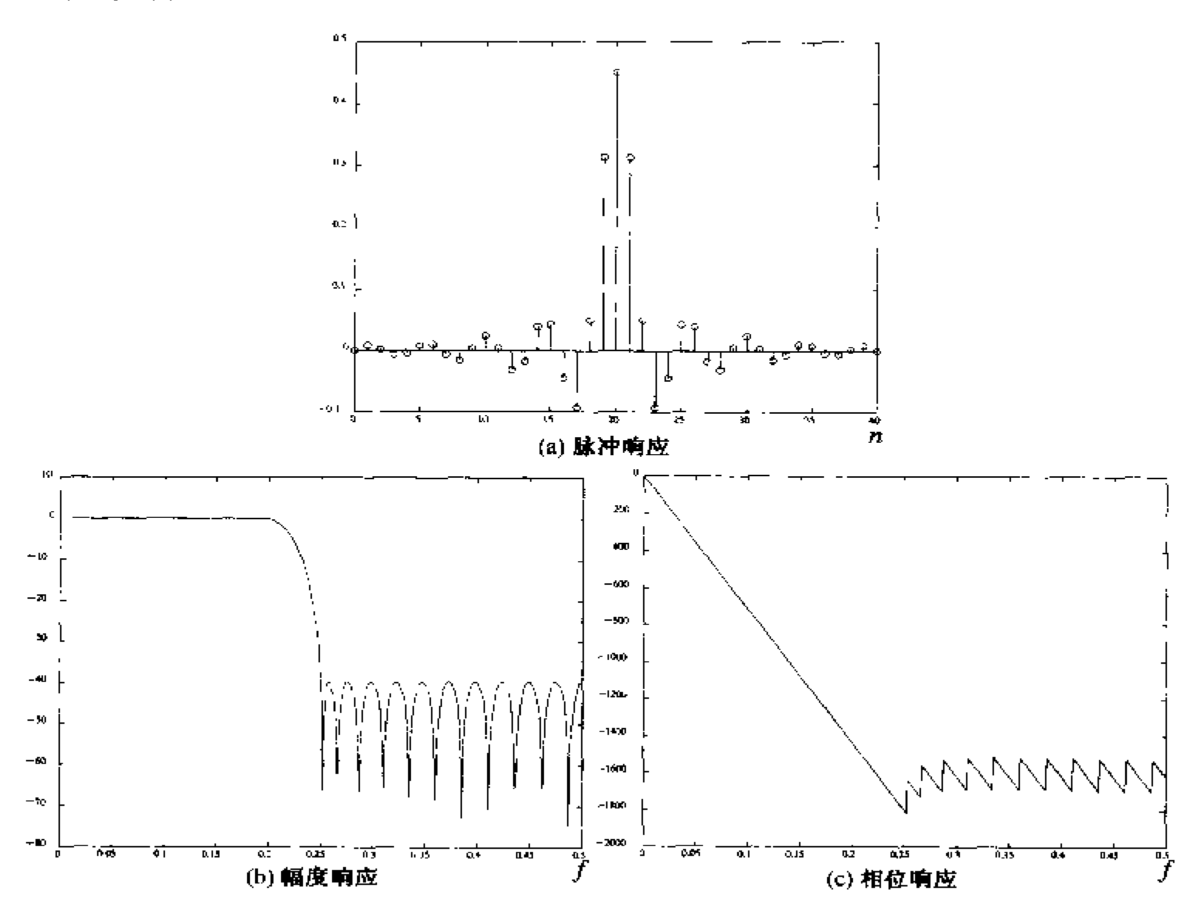


图 6.8　解说题 6.3 中的具有线性相位 FIR 滤波器的脉冲、幅度和相位响应

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 6.3.
echo on
f_cutoff=2000;                % the desired cutoff frequency
f_stopband=2500;              % the actual stopband frequency
fs=10000;                     % the sampling frequency
f1=2*f_cutoff/fs;             % the normalized passband frequency
f2=2*f_stopband/fs;           % the normalized stopband frequency
N=41;                         % This number is found by experiment.
F=[0 f1 f2 1];
M=[1 1 0 0];                  % describes the lowpass filter
B=remez(N-1,F,M);             % returns the FIR tap coefficients
% Plotting command follows.
figure(1);
[H,W]=freqz(B);
H_in_dB=20*log10(abs(H));
plot(W/(2*pi),H_in_dB);
figure(2);
plot(W/(2*pi),(180/pi)*unwrap(angle(H)));
```

```
% Plot of the impulse response follows.
figure(3);
plot(zeros(size([0:N-1])));
hold;
stem([0:N-1],B);
```

解说题

解说题 6.4

设计逼近所期望的信道特性的 FIR 滤波器的另一种方法是基于窗函数法。具体来讲，若期望信道特性是 $C(f), |f| \leqslant W$ 和 $C(f)=0, |f|>W$，那么信道的脉冲响应是：

$$h(t) = \int_{-W}^{W} C(f) e^{j2\pi ft} df \tag{6.3.3}$$

例如，若信道是理想的，则 $C(f)=1, |f| \leqslant W$，所以

$$h(t) = \frac{\sin(2\pi Wt)}{\pi t} \tag{6.3.4}$$

可以通过在 $t=nT_s$ 对 $h(t)$ 采样来实现一个等效的数字滤波器，这里的 T_s 是采样时间间隔，$n=0, \pm 1, \pm 2, \cdots$。试用 $W=2000$ Hz 和 $F_s=1/T_s=10$ kHz 设计一个 FIR 滤波器。

题解

$h(t)$ 的采样 $h_n \equiv h(nT_s)$ 如图 6.9 所示。因为 $\{h_n\}$ 为无限长，可以在某个长度 N 将其截断。这个截断就相当于将 $\{h_n\}$ 乘以矩形窗序列 $w_n=1, |n| \leqslant (N-1)/2$ 和 $w_n=0, |n| \geqslant (N+1)/2$。这个截断后的 FIR 滤波器的脉冲响应 $\{h_n^1 = w_n h_n\}$ 和对应的频率响应如图 6.10 所示 ($N=51$)。注意，截断滤波器在阻带有大的旁瓣，所以这个 FIR 滤波器对所期望的信道特性来说是一个不够好的近似。使用比较平滑的窗函数，如 Hanning 窗或 Hamming 窗，来截断理想信道响应，可以大大降低旁瓣的大小。图 6.11 显示的是使用 $N=51$ 的 Hanning 窗函数，$\{h_n^1 = w_n h_n\}$ 的脉冲响应和频率响应。MATLAB 提供了有关子程序，可以实现几个不同形式的窗函数。

m文件

```
% MATLAB script for Illustrative Problem 6.4.
echo on
Length=101;
Fs=10000;
W=2000;
Ts=1/Fs;
n=-(Length-1)/2:(Length-1)/2;
t=Ts*n;
h=2*W*sinc(2*W*t);
% The rectangular windowed version follows.
N=61;
rec_windowed_h=h((Length-N)/2+1:(Length+N)/2);
% Frequency response of rec_windowed_h follows.
[rec_windowed_H,W1]=freqz(rec_windowed_h,1);
% to normalize the magnitude
rec_windowed_H_in_dB=20*log10(abs(rec_windowed_H)/abs(rec_windowed_H(1)));
% The Hanning windowed version follows.
hanning_window=hanning(N);
hanning_windowed_h=h((Length-N)/2+1:(Length+N)/2).*hanning_window.';
```

```
[hanning_windowed_H,W2]=freqz(hanning_windowed_h,1);
hanning_windowed_H_in_dB=20*log10(abs(hanning_windowed_H)/abs(hanning_windowed_H(1)));
% The plotting commands follow.
```

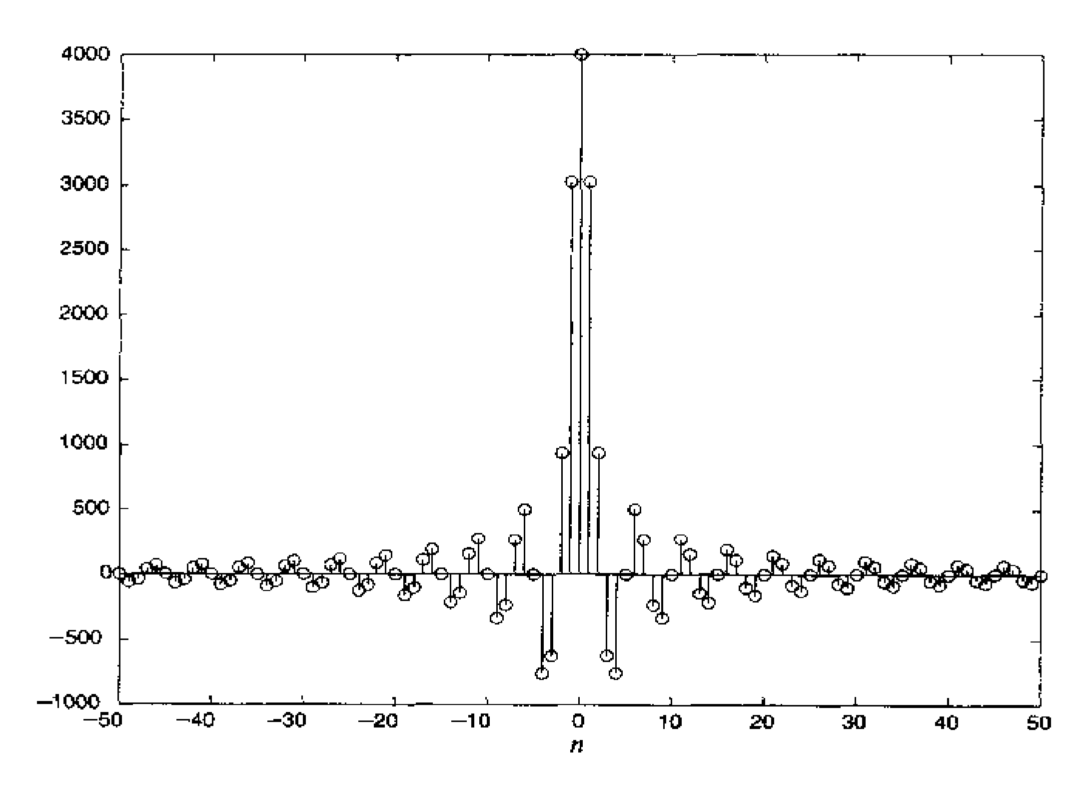


图 6.9　解说题 6.4 中的 $h(n)$ 的样本

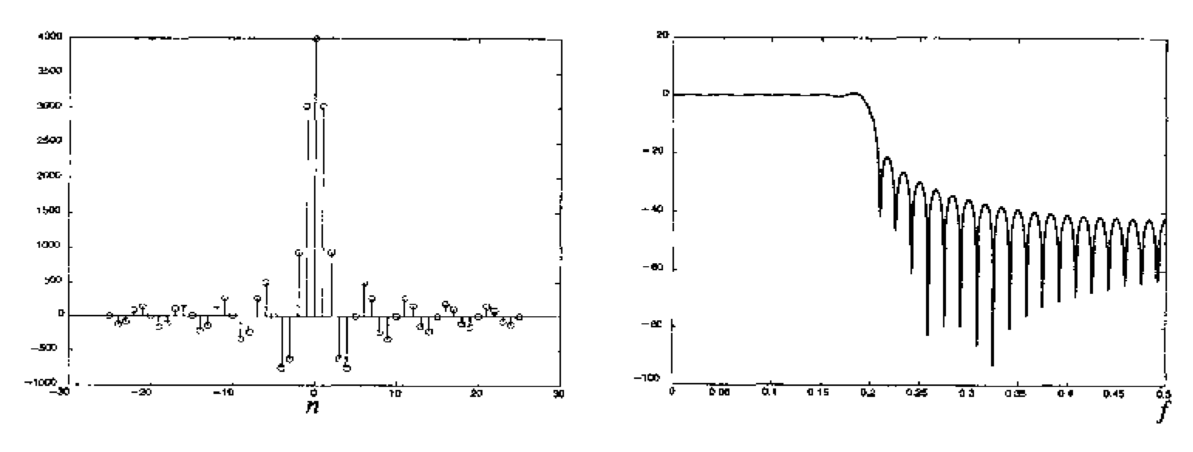


图 6.10　解说题 6.4 中的用矩形窗截断的滤波器的脉冲响应和频率响应

解说题

解说题 6.5

两条路径(多路径)的无线信道可按图 6.12 在时域中建模。它的冲激响应可表示为：

$$c(t,\tau)=b_1(t)\delta(\tau)+b_2(t)\delta(\tau-\tau_d) \tag{6.3.5}$$

其中 $b_1(t)$ 和 $b_2(t)$ 都是随机过程，代表信道的时变传播行为，τ_d 是两条多径分量之间的延时。在计算机上仿真这样的信道。

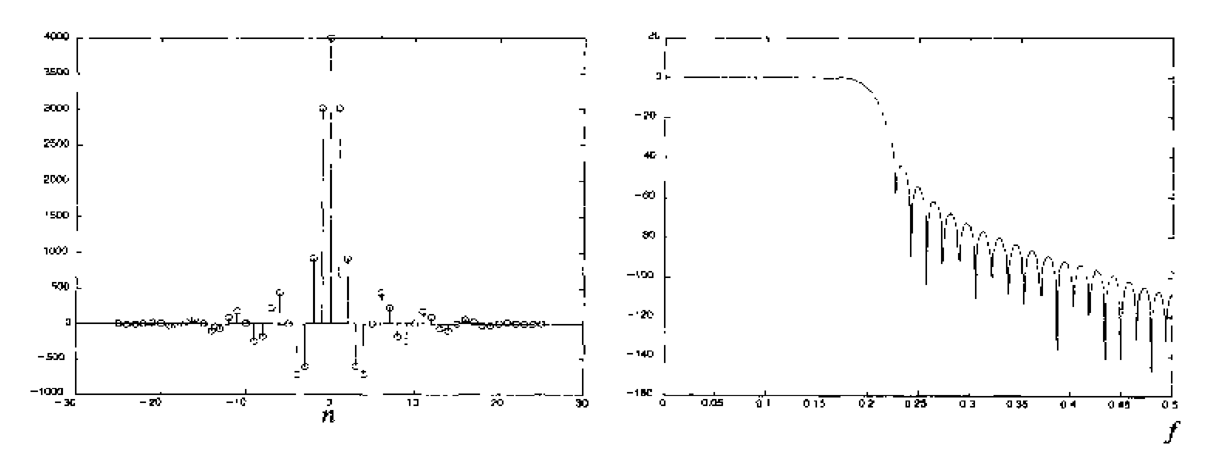


图 6.11　解说题 6.4 中用 Hanning 窗截断的滤波器的脉冲响应和频率响应

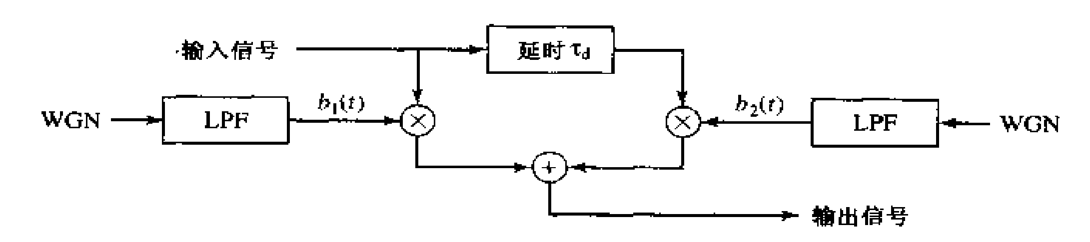


图 6.12　两条路径的无线信道模型

题　解

将高斯白噪声过程通过低通滤波器(LPF)产生高斯随机过程来仿真 $b_1(t)$ 和 $b_2(t)$，在离散时间，可以相对简单地用高斯白噪声(WGN)序列激励数字 IIR 滤波器来实现。例如，具有两个相同极点的简单低通滤波器用 z 变换可以表示为：

$$H(z)=\frac{(1-p)^2}{(1-pz^{-1})^2}=\frac{(1-p)^2}{1-2pz^{-1}+p^2z^{-2}} \tag{6.3.6}$$

或者对应的差分方程是：

$$b_n=2pb_{n-1}-p^2b_{n-2}+(1-p)^2w_n \tag{6.3.7}$$

其中，$\{w_n\}$ 是输入 WGN 序列，$\{b_n\}$ 是输出序列，$p(0<p<1)$ 是极点的位置。极点的位置控制该滤波器的带宽，从而也就是 $\{b_n\}$ 的变化速率。当 p 靠近 1(即接近单位圆)时，滤波器的带宽就窄；而当 p 靠近 0 时，带宽就宽。所以，当 p 在 z 平面内接近单位圆时，滤波器的输出序列的变化就比当 p 接近原点时更慢一些。

图 6.13 显示的是当 $p=0.99$ 时，将统计独立的 WGN 序列通过该滤波器所产生的输出序列 $\{b_{1n}\}$ 和 $\{b_{2n}\}$，同时也显示了离散时间信道脉冲响应 $\{c_n\}$

$$c_n=b_{1,n}+b_{2,n-d} \tag{6.3.8}$$

样本延时为 $d=5$。图 6.14 是 $p=0.9$ 时序列 $\{b_{1n}\}$，$\{b_{2n}\}$ 和 $\{c_n\}$ 的图。

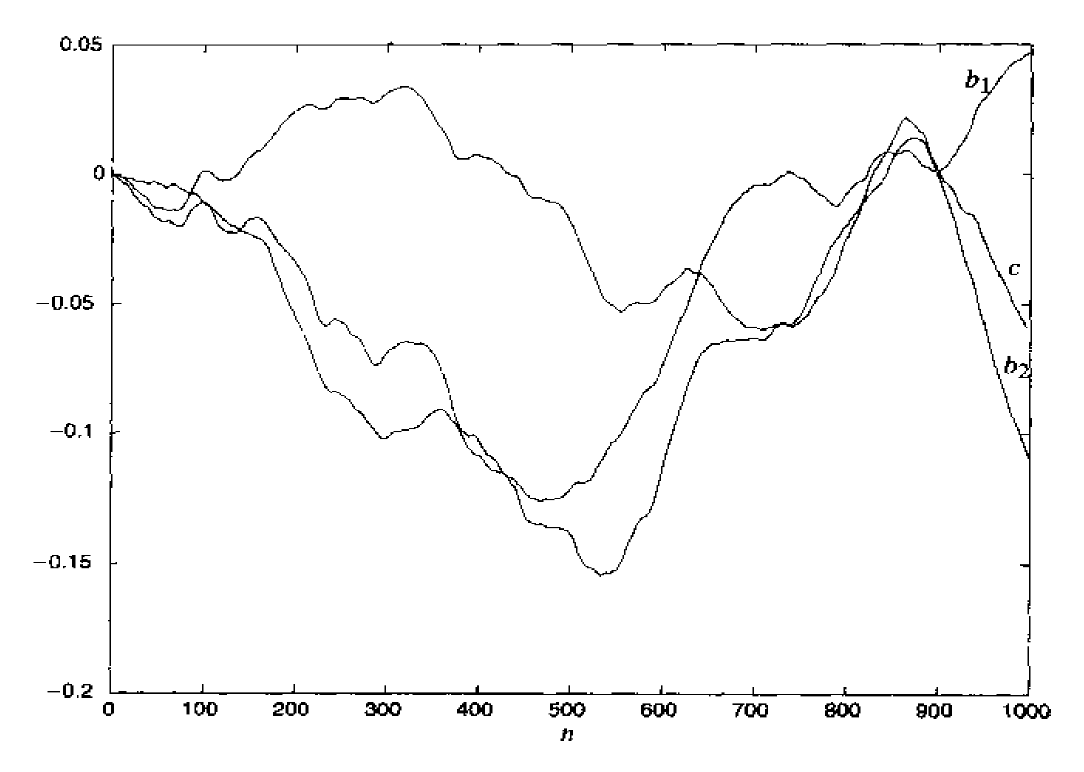


图 6.13　当 $p=0.99$ 时，低通滤波器的输出序列 b_{1n} 和 b_{2n} 及其所产生的 c_n

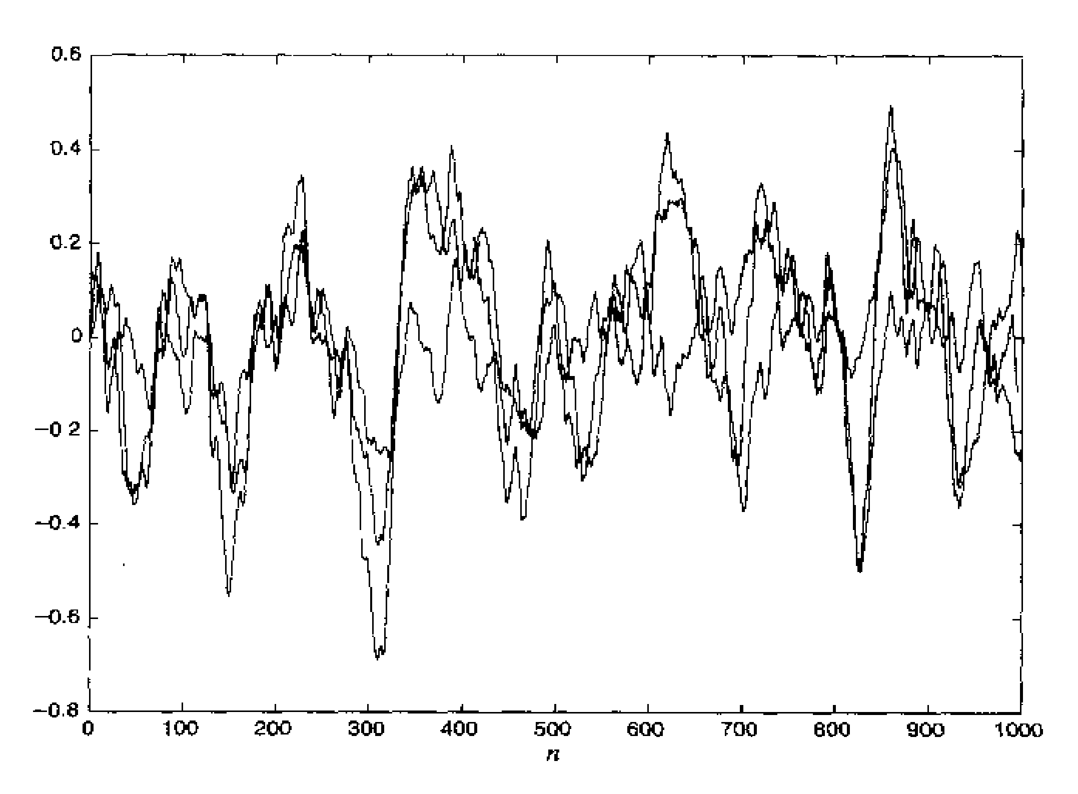


图 6.14　当 $p=0.9$ 时的输出序列 b_{1n}，b_{2n} 和 c_n（在 $n=1000$ 时，由上至下分别为 b_{1n}，c_n 和 b_{2n}）

6.4 码间干扰的特性

在数字通信系统中,信道失真会导致码间干扰(ISI)。这一节要给出表征 ISI 的一种模型。为简单起见,假定传输的信号是基带 PAM 信号。然而,这一论述很容易推广到下一章要讨论的载波(线性)调制信号中。

传输的 PAM 信号表示为:

$$s(t) = \sum_{n=0}^{\infty} a_n g(t - nT) \tag{6.4.1}$$

其中,$g(t)$是要选择的基本脉冲形状,用于控制传输信号的频谱特性。$\{a_n\}$是由 M 个点组成的信号星座图中选取的传输信息符号的序列。T 是信号区间($1/T$ 就是符号率)。

在一个基带信道上传输信号 $s(t)$,该基带信道可用频率响应 $C(f)$表征。这样,接收信号可表示成:

$$r(t) = \sum_{n=0}^{\infty} a_n h(t - nT) + w(t) \tag{6.4.2}$$

其中,$h(t)=g(t)*c(t)$,$c(t)$是信道的冲激响应,$*$代表卷积,而 $w(t)$代表的是在信道中的加性噪声。为了表征 ISI,假设接收信号通过一个接收滤波器,然后以每秒 $1/T$ 样本的采样率进行采样。一般来说,在接收端最佳滤波器是与接收信号脉冲 $h(t)$相匹配的,所以这个滤波器的频率响应是 $H^*(f)$。滤波器的输出表示为:

$$y(t) = \sum_{n=0}^{\infty} a_n x(t - nT) + v(t) \tag{6.4.3}$$

其中,$x(t)$是接收滤波器的信号脉冲响应,即 $X(f)=H(f)H^*(f)=|H(f)|^2$,$v(t)$是接收滤波器对噪声 $w(t)$的响应。现在,如果 $y(t)$在时刻 $t=kT$,$k=0,1,2,\cdots$被采样,就有

$$y(kT) = \sum_{n=0}^{\infty} a_n x(kT - nT) + v(kT)$$

$$y_k = \sum_{n=0}^{\infty} a_n x_{k-n} + v_k, \quad k = 0,1,\cdots \tag{6.4.4}$$

样本值$\{y_k\}$可以表示为:

$$y_k = x_0\left(a_k + \frac{1}{x_0}\sum_{\substack{n=0 \\ n\neq k}}^{\infty} a_n x_{k-n}\right) + v_k, k = 0,1,\cdots \tag{6.4.5}$$

x_0 是任意加权因子,为了方便,我们将其置为 1,那么

$$y_k = a_k + \sum_{\substack{n=0 \\ n\neq k}}^{\infty} a_n x_{k-n} + v_k \tag{6.4.6}$$

a_k 项就代表了在第 k 个采样瞬时所期望的信息符号,而

$$\sum_{\substack{n=0 \\ n\neq k}}^{\infty} a_n x_{k-n} \tag{6.4.7}$$

项就代表了码间干扰 ISI,而 v_k 是在第 k 个采样瞬时的加性噪声。

在一个数字通信系统中,ISI 和噪声的量都能在示波器上观察到。对 PAM 信号可以用水平扫描速率 $1/T$,在垂直输入上展示接收信号 $y(t)$。所得出的示波器上的图称为**眼图**,因为它

与人的眼睛很相像。例如，图 6.15 显示的是 2 电平和 4 电平 PAM 调制的眼图。ISI 的影响是导致眼睛闭合，因此降低了因加性噪声引起误码的裕度。图 6.16 用图解方法说明了 ISI 在减小一个 2 电平眼睛开启方面的影响。值得注意的是，码间干扰使得过零点的位置产生偏差，从而减小了眼图的开启度，因此会导致系统对同步误差有更高的灵敏度。

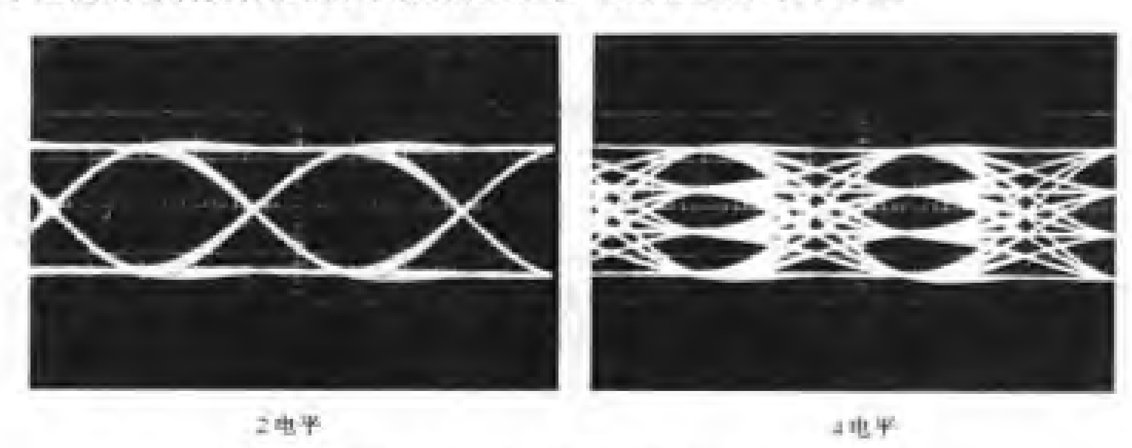


图 6.15　2 电平和 4 电平幅度键控(或 PAM)眼图举例

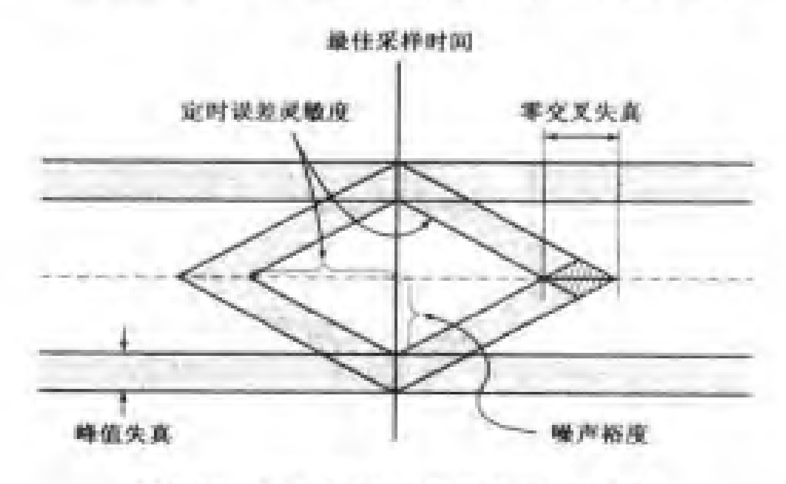


图 6.16　码间干扰在眼图开启方面的影响

解说题

解说题 6.6

本题要考虑码间干扰(ISI)对两个信道接收到的信号序列$\{y_k\}$的影响，这两个信道产生的序列$\{x_k\}$如下所示：

信道 1

$$x_n = \begin{cases} 1, & n = 0 \\ -0.25, & n = \pm 1 \\ 0.1, & n = \pm 2 \\ 0, & \text{其余 } n \end{cases}$$

信道 2

$$x_n = \begin{cases} 1, & n = 0 \\ 0.5, & n = \pm 1 \\ -0.2, & n = \pm 2 \\ 0, & \text{其余 } n \end{cases}$$

注意,这个ISI局限在欲传输信号两边的两个符号上。因此,发送端和接收端滤波器的级联和在采样瞬时的信道就可以用如图6.17所示的这个等效**离散时间FIR信道滤波器**来表示。现在,假设传输信号序列是2电平的,即$\{a_n = \pm 1\}$。那么,对于信道1而言,在没有噪声的情况下接收到的信号序列$\{y_k\}$如图6.18(a)所示,而在有方差$\sigma^2 = 0.1$的加性高斯白噪声的情况下,接收到的信号序列如图6.18(b)所示。可以注意到,在没有噪声时,仅由ISI并不产生在检测器上的差错(这个检测器将接收到的信号序列$\{y_n\}$与阈值为零进行比较),所以这时眼图是开启的。然而,当加性噪声足够大时,就会发生差错。

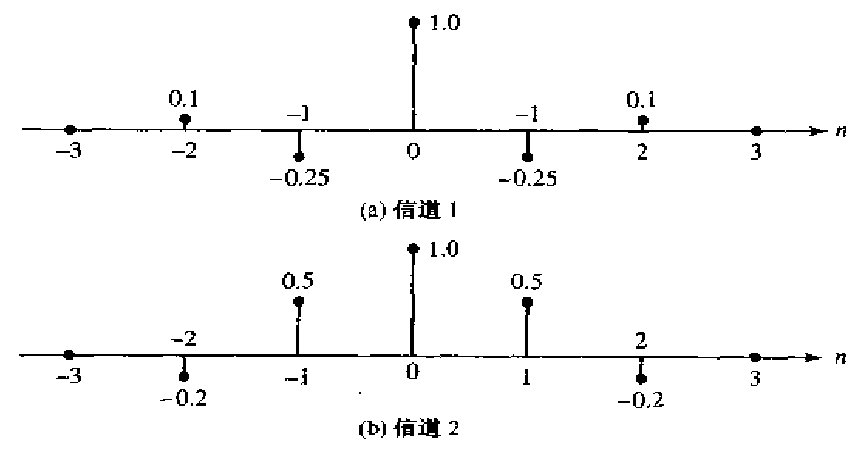


图6.17 具有ISI的FIR信道模型

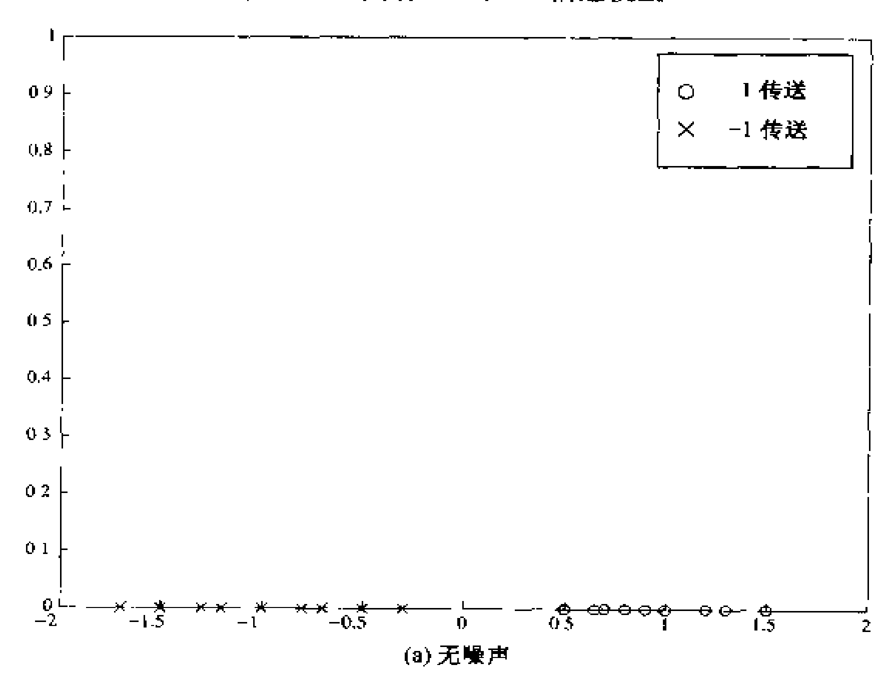
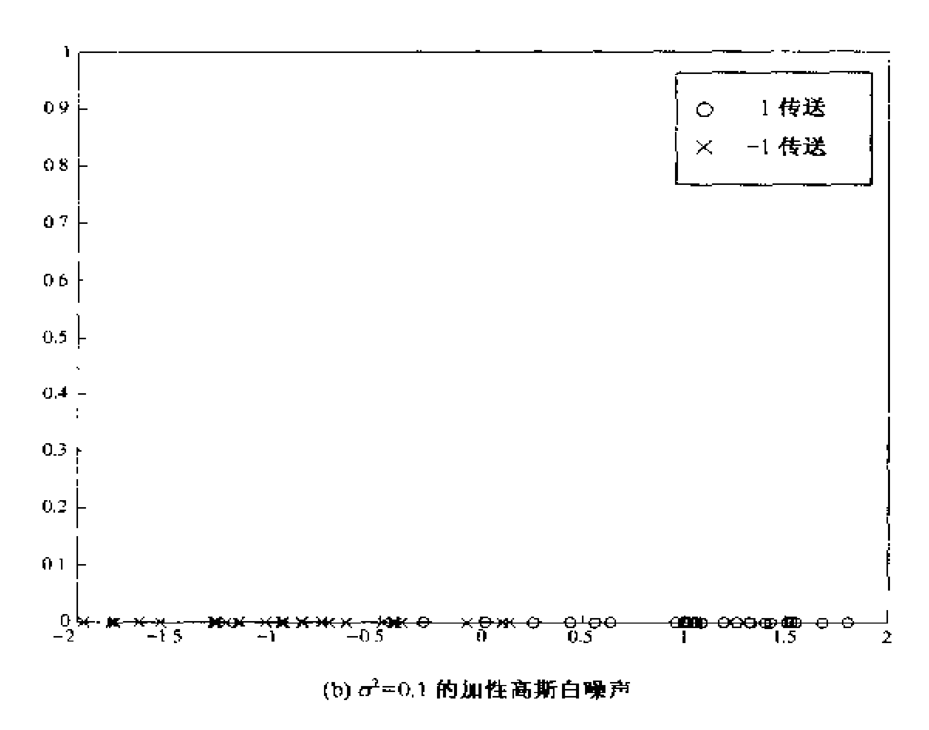


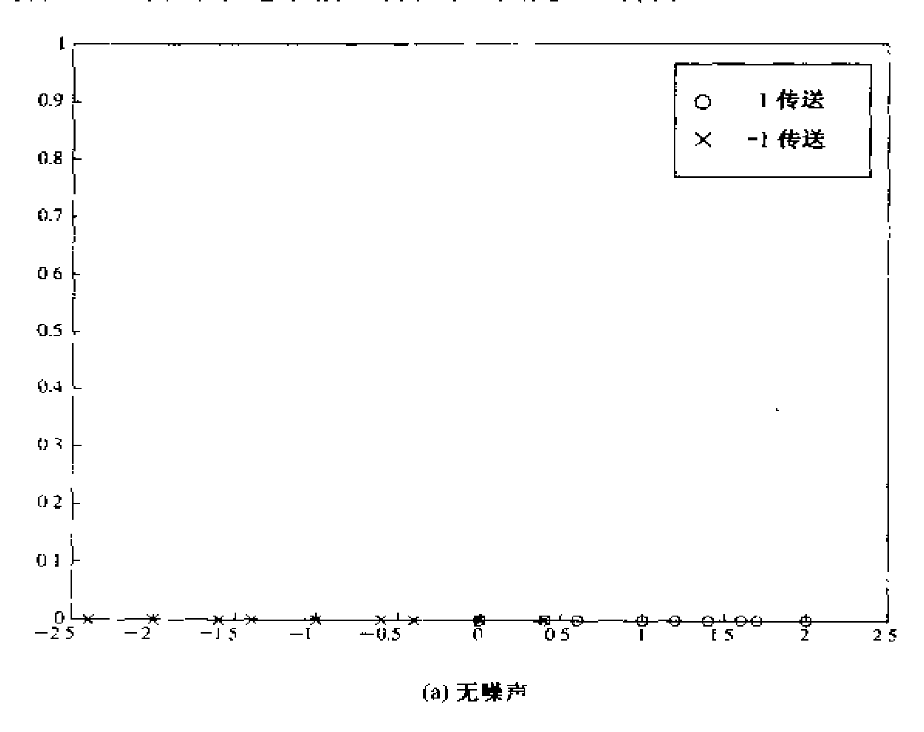
图6.18 有和没有AWGN时,信道模型1的输出



(b) $\sigma^2=0.1$ 的加性高斯白噪声

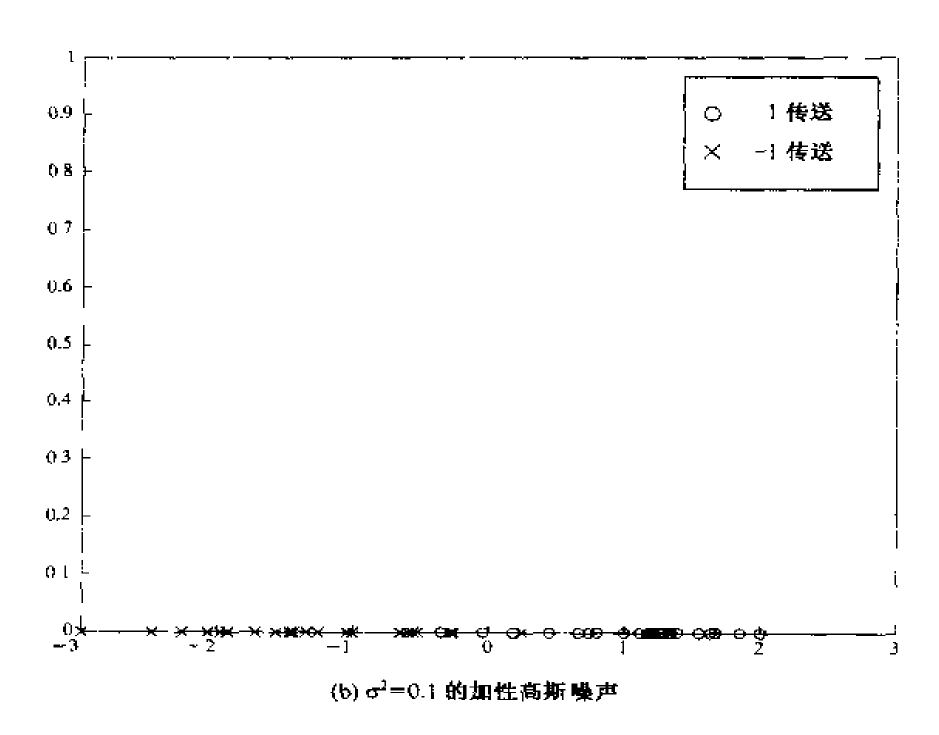
图 6.18(续) 有和没有 AWGN 时,信道模型 1 的输出

在信道 2 的情况下,无噪声和有噪声($\sigma^2=0.1$)时的序列$\{y_n\}$如图 6.19 所示。现在可以看到,当检测器将接收序列$\{y_n\}$与阈值为零进行比较后,在检测器上 ISI 会产生差错,即使在无噪声时也是这样。因此,对于这个信道特性,眼图完全闭合。



(a) 无噪声

图 6.19 有和没有 AWGN 时,信道模型 2 的输出



(b) $\sigma^2=0.1$ 的加性高斯噪声

图 6.19(续) 有和没有 AWGN 时,信道模型 2 的输出

6.5 带限信道的通信系统设计

这一节要考虑发送端和接收端滤波器的设计问题,这些滤波器对基带带限信道是适用的。要考虑两种情况:第一种情况是基于发送和接收滤波器产生零 ISI 的设计;第二种情况是基于发送和接收滤波器具有某个给定(预定)的 ISI 值的设计。因此,第二种设计方法会导致某个可控制的 ISI 值,所对应的传输信号称为**部分响应信号**。在两种情况下都假定信道是理想的;也就是说,在信道带宽 W 内,$A(f)$和 $\tau(f)$都是常数。为简单起见,设 $A(f)=1$ 和 $\tau(f)=0$。

6.5.1 对于零 ISI 的信号设计

具有零 ISI 的带限信号设计是大约 70 年前 Nyquist 考虑过的一个问题。他证实:一个信号 $x(t)$具有零 ISI 的必要和充分条件是:

$$x(nT)=\begin{cases}1, & n=0\\0, & n\neq 0\end{cases} \tag{6.5.1}$$

它的傅里叶变换 $X(f)$满足

$$\sum_{m=-\infty}^{\infty} X\left(f+\frac{m}{T}\right)=T \tag{6.5.2}$$

其中 $1/T$ 是符号率。

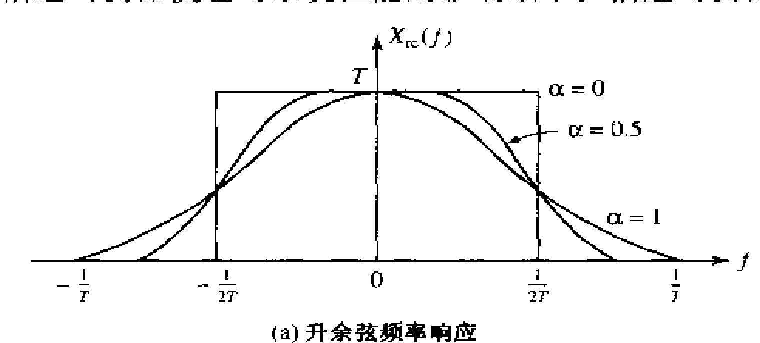
一般来说,很多信号都设计成具有这个性质。在实际中最常用的一种信号是具有升余弦频率响应特性的信号,定义为:

$$X_{rc}(f)=\begin{cases}T, & 0\leqslant |f|\leqslant \dfrac{1-\alpha}{2T}\\ \dfrac{T}{2}\left[1+\cos\dfrac{\pi T}{\alpha}\left(|f|-\dfrac{1-\alpha}{2T}\right)\right], & \dfrac{1-\alpha}{2T}<|f|\leqslant \dfrac{1+\alpha}{2T}\\ 0, & |f|>\dfrac{1+\alpha}{2T}\end{cases} \tag{6.5.3}$$

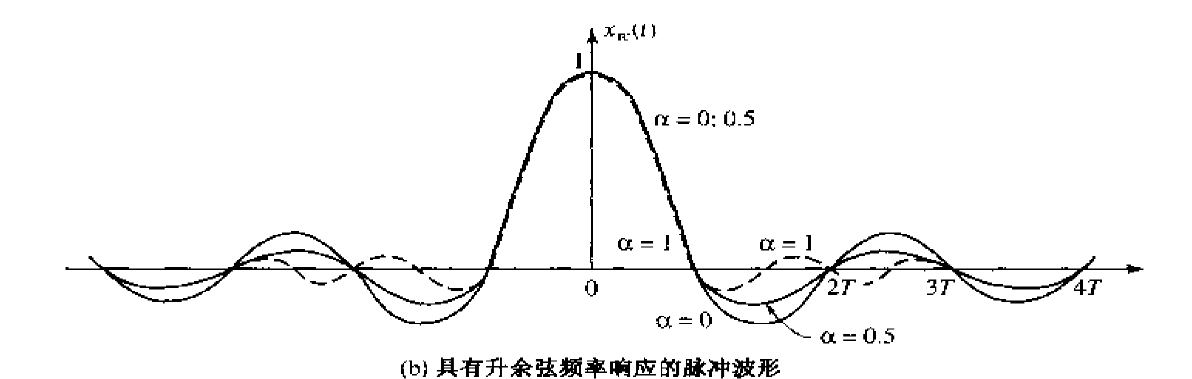
其中，α 称为**滚降系数**，它的取值为 $0\leqslant\alpha\leqslant 1$，$1/T$ 是符号率。$\alpha=0$，$\alpha=1/2$ 和 $\alpha=1$ 的频率响应 $X_{rc}(f)$ 如图6.20(a)所示。注意，当 $\alpha=0$ 时，$X_{rc}(f)$ 就变成一个理想的、带宽为 $1/(2T)$ 的物理上不可实现的矩形频率响应。频率 $1/(2T)$ 称为**奈奎斯特频率**。对于 $\alpha>0$，被期望信号 $X_{rc}(f)$ 占据的超过奈奎斯特频率 $1/(2T)$ 的带宽称为**过量带宽**(excess bandwidth)，通常将它表示为奈奎斯特频率的一个百分数。例如，当 $\alpha=1/2$ 时，过量带宽是50%；当 $\alpha=1$ 时，过量带宽是100%。具有升余弦频谱的信号脉冲 $x_{rc}(t)$ 是：

$$x_{rc}(t)=\frac{\sin(\pi t/T)}{\pi t/T}\frac{\cos(\pi\alpha t/T)}{1-4\alpha^2t^2/T^2} \tag{6.5.4}$$

图6.20(b)给出了 $\alpha=0$，$\alpha=1/2$ 和 $\alpha=1$ 时的 $x_{rc}(t)$。因为 $x_{rc}(f)$ 满足式(6.5.2)，所以有 $x_{rc}(t)=1$，$t=0$ 和 $x_{rc}(t)=0$，$t=kT$，$k=\pm1,\pm2,\cdots$。这样，在采样瞬时 $t=kT$，$k\neq0$，在不存在信道失真的情况下，不存在从邻近符号来的ISI。然而，在有信道失真时，由式(6.4.7)给出的ISI不再是零，需要用一个信道均衡器使它对系统性能的影响最小。信道均衡器在6.6节中讨论。



(a) 升余弦频率响应



(b) 具有升余弦频率响应的脉冲波形

图6.20　升余弦函数

在理想信道中，发送和接收滤波器联合设计成在所要求的采样瞬时 $t=nT$ 具有零ISI。因此，若 $G_T(f)$ 是发送滤波器的频率响应，而 $G_R(f)$ 是接收滤波器的频率响应，那么乘积 $G_T(f)G_R(f)$（两个滤波器级联）就设计为产生零ISI。例如，如果该乘积 $G_T(f)G_R(f)$ 选为：

$$G_T(f)G_R(f)=X_{rc}(f) \tag{6.5.5}$$

其中，$X_{rc}(f)$是升余弦频率响应特性，那么在采样时刻 $t=nT$，ISI 是零。

解说题

解说题 6.7

设计一个发送和接收滤波器 $G_T(f)$和 $G_R(f)$的数字实现，使它们的乘积满足式(6.5.5)，并且 $G_R(f)$是 $G_T(f)$的匹配滤波器。

题　解

以数字形式设计和实现发送和接收滤波器的最简单方法是采用具有线性相移(对称脉冲响应)的 FIR 滤波器。所期望的幅度响应是：

$$|G_T(f)|=|G_R(f)|=\sqrt{X_{rc}(f)} \tag{6.5.6}$$

其中，$X_{rc}(f)$由式(6.5.3)给出。频率响应与数字滤波器脉冲响应的关系是：

$$G_T(f)=\sum_{m=-(N-1)/2}^{(N-1)/2} g_T(n)e^{-j2\pi fnT_s} \tag{6.5.7}$$

其中，T_s 是采样间隔，N 是滤波器长度，注意 N 是奇数。因为 $G_T(f)$是带限的，可以选采样频率 F_s 至少是 $2/T$。现选择 F_s 为

$$F_s=\frac{1}{T_s}=\frac{4}{T}$$

或者等效为 $T_s=T/4$。所以，折叠频率是 $F_s/2=2/T$。因为 $G_T(f)=\sqrt{X_{rc}(f)}$，所以在频域以 $\Delta f=F_s/N$ 等频率间隔对 $X_{rc}(f)$采样，有

$$\sqrt{X_{rc}(m\Delta f)}=\sqrt{X_{rc}\left(\frac{mF_s}{N}\right)}=\sum_{n=-(N-1)/2}^{(N-1)/2} g_T(n)e^{-j2\pi mn/N} \tag{6.5.8}$$

它的逆变换关系是：

$$g_T(n)=\sum_{m=-(N-1)/2}^{(N-1)/2}\sqrt{X_{rc}\left(\frac{4m}{NT}\right)}e^{j2\pi mn/N},\qquad n=0,\pm1,\cdots,\pm\frac{N-1}{2} \tag{6.5.9}$$

由于 $g_T(n)$是对称的，因此将 $g_T(n)$延时$(N-1)/2$ 个样本就得到了期望的线性相位发送滤波器的脉冲响应。

计算本题的 MATLAB 脚本如下所示。

m 文件

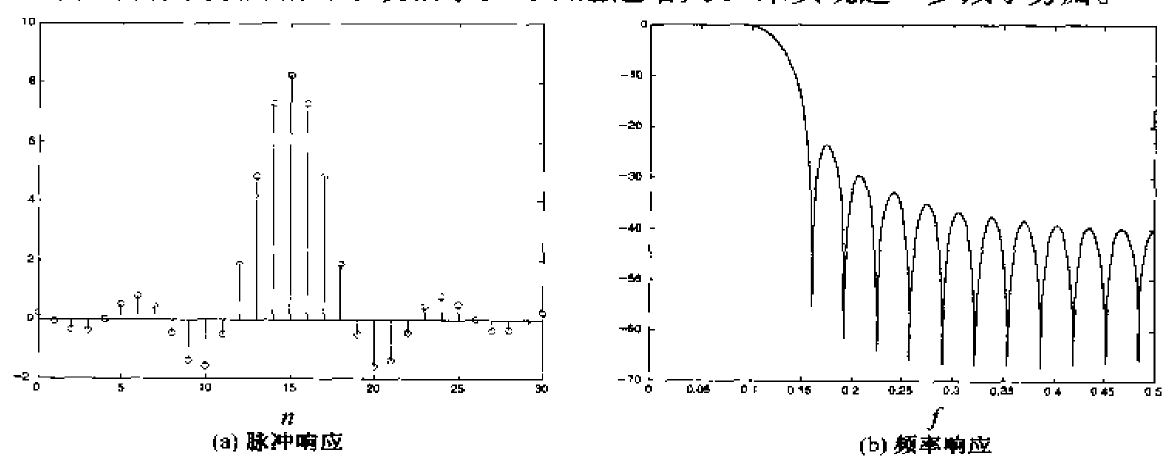
```
% MATLAB script for Illustrative Problem 6.7.
echo on
N=31;
T=1;
alpha=1/4;
n=-(N-1)/2:(N-1)/2;                    % the indices for g_T
% The expression for g_T is obtained next.
for i=1:length(n),
  g_T(i)=0;
  for m=-(N-1)/2:(N-1)/2,
    g_T(i)=g_T(i)+sqrt(xrc(4*m/(N*T),alpha,T))*exp(j*2*pi*m*n(i)/N);
```

```
    echo off ;
  end;
end;
echo on ;
g_T=real(g_T) ; % The imaginary part is due to the finite machine precision.
% Derive g_T(n-(N-1)/2).
n2=0:N-1;
% Get the frequency response characteristics.
[G_T,W]=freqz(g_T,1);
% normalized magnitude response
magG_T_in_dB=20*log10(abs(G_T)/max(abs(G_T)));
% impulse response of the cascade of the transmitter and the receiver filters
g_R=g_T;
imp_resp_of_cascade=conv(g_R,g_T);
% Plotting commands follow.
```

m 文件

```
function [y] = xrc(f,alpha,T);
% [y]=xrc(f,alpha,T)
%               Evaluates the expression Xrc(f). The parameters alpha and T
%               must also be given as inputs to the function.
if (abs(f) > ((1+alpha)/(2*T))),
  y=0;
elseif (abs(f) > ((1-alpha)/(2*T))),
  y=(T/2)*(1+cos((pi*T/alpha)*(abs(f)-(1-alpha)/(2*T))));
else
  y=T;
end;
```

图 6.21(a)给出了当 $\alpha=\frac{1}{4}$和 $N=31$ 时的 $g_T(n-\frac{N-1}{2})$, $n=0,1,\cdots,N-1$,图 6.21(b)是相应的频率响应特性。注意,因为滤波器为有限长,所以频率响应在 $|f|\geqslant(1+\alpha)/T$ 不再是零。然而,在频谱中旁瓣相对来说很小。可以通过增大 N 来实现进一步减小旁瓣。



(a) 脉冲响应　　(b) 频率响应

图 6.21　发送端截断离散时间 FIR 滤波器的脉冲和频率响应

最后,在图 6.22 中给出了发送和接收 FIR 滤波器级联后的脉冲响应。这个响应与在 $F_s=4/T$ 的采样频率下对 $x_{rc}(t)$采样所得到的理想脉冲响应是可以比拟的。

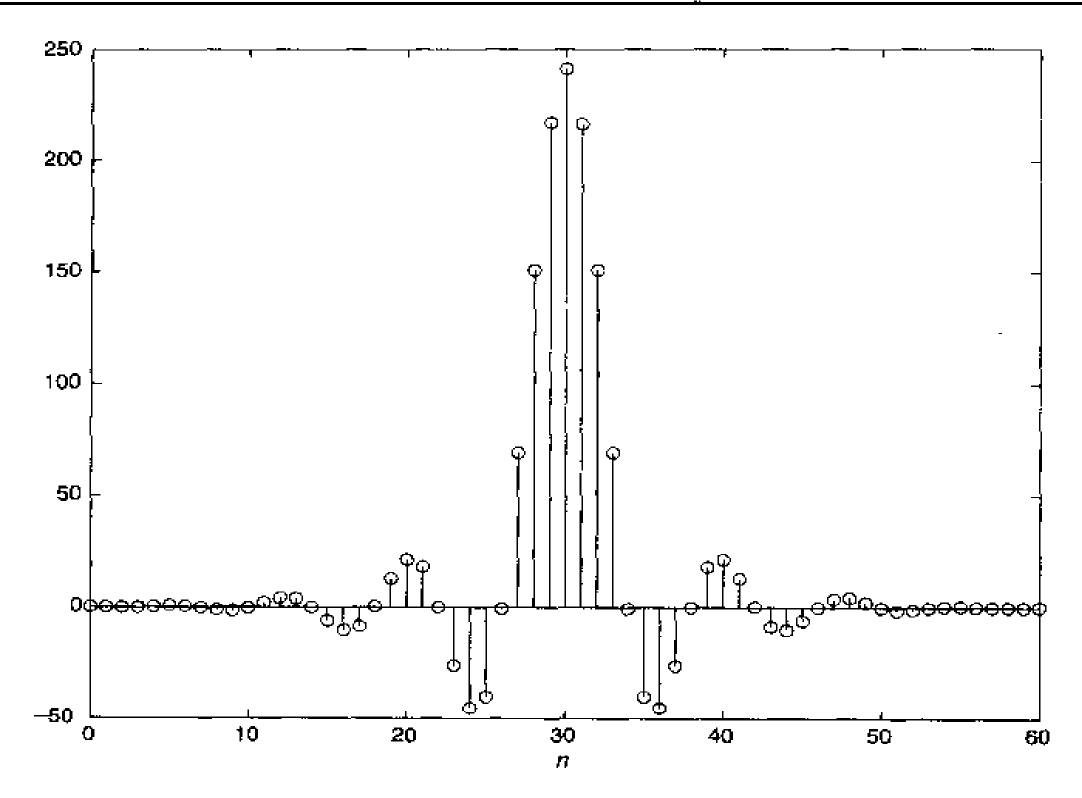


图 6.22 发送滤波器与在接收端匹配滤波器级联的脉冲响应

6.5.2 可控制 ISI 的信号设计

正如我们在零 ISI 的信号设计讨论中已经看到的,可以用具有过量带宽的传输滤波器来实现实际带限信道的发送和接收滤波器。另一方面,设想选择放松零 ISI 的条件而实现带宽 $W=1/(2T)$的符号传输,也就是说不用过量带宽。通过允许某个可控制的 ISI 量,就能实现每秒 $2W$ 个符号的速率。

我们已经看到,零 ISI 的条件是 $x(nT)=0, n\neq 0$。然而,假定设计的带限信号在一个时间瞬间具有可控制的 ISI,这意味着允许在样本$\{x(nT)\}$中有一个额外的非零值。由于引入的 ISI 是确定的,或者说"可控制的",所以在接收端就能将它计入,这一点稍后将进行讨论。

一般来说,一个带限到 W Hz 的信号 $x(t)$,即

$$X(f)=0, \qquad |f|>W \tag{6.5.10}$$

可以表示成:

$$x(t)=\sum_{n=-\infty}^{\infty} x\left(\frac{n}{2W}\right)\frac{\sin\{2\pi W[t-n/(2W)]\}}{2\pi W[t-n/(2W)]} \tag{6.5.11}$$

上式是从带限信号的采样定理中得出的。带限信号的频谱是:

$$X(f)=\int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}\,dt = \begin{cases}\dfrac{1}{2W}\displaystyle\sum_{n=-\infty}^{\infty} x\left(\frac{n}{2W}\right)e^{-j n\pi f/W}, & |f|\leqslant W\\ 0, & |f|>W\end{cases} \tag{6.5.12}$$

导致物理上可实现的发送和接收滤波器的一种特殊情况样本由下式给出:

$$x\left(\frac{n}{2W}\right)\equiv x(nT)=\begin{cases}1, & n=0,1\\ 0, & \text{其余 } n\end{cases} \tag{6.5.13}$$

相应的信号频谱是：

$$X(f)=\begin{cases}\frac{1}{2W}\left[1+e^{-j n\pi f/W}\right], & |f|<W\\ 0, & \text{其余 } f\end{cases}$$

$$=\begin{cases}\frac{1}{W}e^{-j2\pi f/2W}\cos\left(\frac{\pi f}{2W}\right), & |f|<W\\ 0, & \text{其余 } f\end{cases} \tag{6.5.14}$$

因此，$x(t)$由下式给出：

$$x(t)=\text{sinc}(2Wt)+\text{sinc}(2Wt-1) \tag{6.5.15}$$

其中，$\text{sinc}(t)=\sin(\pi t)/(\pi t)$。这个脉冲称为**双二进制信号脉冲**(duobinary signal pulse)，它的波形及幅度谱如图 6.23 所示。

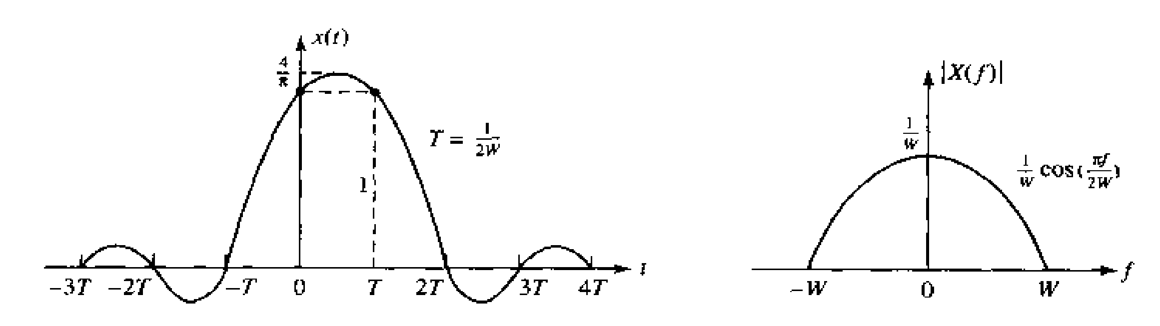


图 6.23　双二进制信号脉冲及其频谱

注意，这个频谱平滑衰减到零，这就意味着能够设计出物理上可实现的滤波器，它能非常近似这个频谱，从而可实现 $2W$ 的符号速率。

导致物理上可实现的发送和接收滤波器的另一种特殊情况样本由下式给出：

$$x\left(\frac{n}{2W}\right)=x(nT)=\begin{cases}1, & n=1\\ -1, & n=-1\\ 0, & \text{其余 } n\end{cases} \tag{6.5.16}$$

相应的脉冲 $x(t)$为：

$$x(t)=\text{sinc}(2Wt+1)-\text{sinc}(2Wt-1) \tag{6.5.17}$$

它的频谱是：

$$X(f)=\begin{cases}\frac{1}{2W}\left(e^{j\pi f/W}-e^{-j\pi f/W}\right)=\frac{j}{W}\sin\left(\frac{\pi f}{W}\right), & |f|\leqslant W\\ 0, & |f|>W\end{cases} \tag{6.5.18}$$

这个脉冲和它的幅度谱如图 6.24 所示，称为**修正双二进制信号脉冲**(modified duobinary signal pulse)。注意，这个信号的频谱在 $f=0$ 有零值是很有意思的，对于欲在一条不通过直流分量的信道上传输时，这个信号很合适。

通过选择样本$\{x(n/(2W))\}$中不同的值和多于两个非零样本的方法，还能得到其他有趣的和物理可实现的滤波器特性。但是，随着选取更多的非零样本，解决可控 ISI 值的问题就会变得更为复杂和不切实际。

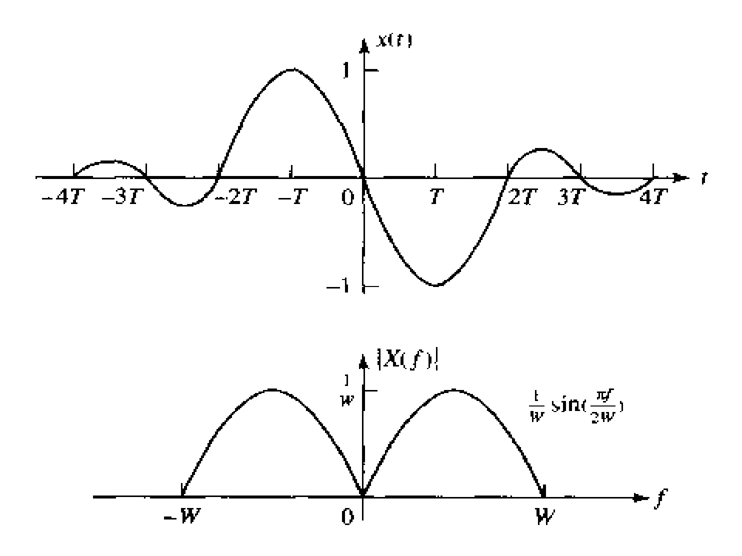


图 6.24　修正双二进制信号脉冲及其频谱

当采用从样本$\{x(n/(2W))\}$集合中选取的两个或更多个非零样本有目的地引入可控的ISI时,所得到的信号称为**部分响应信号**。这样得到的信号脉冲就允许在$2W$样本每秒的奈奎斯特速率下传输信息符号,因此与升余弦信号脉冲相比可获得较大的带宽效率。

解说题

解说题 6.8

设计一个发送和接收滤波器$G_T(f)$和$G_R(f)$的数字实现,使得它们的乘积等于一个双二进制脉冲的频谱,并且$G_R(f)$是$G_T(f)$的匹配滤波器。

题　解

为了满足频域指标,应该有:

$$|G_T(f)||G_R(f)| = \begin{cases} \dfrac{1}{W}\cos\left(\dfrac{\pi f}{2W}\right), & |f| \leqslant W \\ 0, & |f| > W \end{cases} \tag{6.5.19}$$

所以有:

$$|G_T(f)| = \begin{cases} \sqrt{\dfrac{1}{W}\cos\left(\dfrac{\pi f}{2W}\right)}, & |f| \leqslant W \\ 0, & |f| > W \end{cases} \tag{6.5.20}$$

现在,按照与解说题6.7同样的方式求发送和接收滤波器的FIR实现的脉冲响应,用$W = 1/(2T)$和$F_s = 4/T$,可得

$$g_T(n) = \sum_{m=-(N-1)/2}^{(N-1)/2} \left| G_T\left(\frac{4m}{NT}\right) \right| e^{j2\pi mn/N}, \quad n = 0, \pm 1, \pm 2, \cdots, \pm \frac{N-1}{2} \tag{6.5.21}$$

以及$g_R(n) = g_T(n)$。

计算本题的MATLAB脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 6.8.
echo on
N=31;
T=1;
W=1/(2*T);
n=-(N-1)/2:(N-1)/2;                    % the indices for g_T
% The expression for g_T is obtained next.
for i=1:length(n),
  g_T(i)=0;
  for m=-(N-1)/2:(N-1)/2,
    if ( abs((4*m)/(N*T)) <= W ),
      g_T(i)=g_T(i)+sqrt((1/W)*cos((2*pi*m)/(N*T*W)))*exp(j*2*pi*m*n(i)/N);
    end;
    echo off ;
  end;
end;
echo on ;
g_T=real(g_T) ; % The imaginary part is due to the finite machine precision.
% Obtain g_T(n-(N-1)/2).
n2=0:N-1;
% Obtain the frequency response characteristics.
[G_T,W]=freqz(g_T,1);
% normalized magnitude response
magG_T_in_dB=20*log10(abs(G_T)/max(abs(G_T)));
% impulse response of the cascade of the transmitter and the receiver filters
g_R=g_T;
imp_resp_of_cascade=conv(g_R,g_T);
% Plotting commands follow.
```

图 6.25(a)给出了 $N=31$ 时的 $g_T\left(n-\frac{N-1}{2}\right)$，$n=0,1,\cdots,N-1$，相应的频率响应特性如图 6.25(b)所示。可以注意到，由于这个数字滤波器为有限长的，所以频率响应特性在 $|f|>W$ 时不再是零。然而，在频谱中的旁瓣相对很小。最后，在图 6.26 中给出了发送和接收 FIR 滤波器级联的脉冲响应，这个脉冲响应与以 $F_s=4/T=8W$ 采样率对由式(6.5.17)给出的 $x(t)$ 采样得到的理想脉冲响应是可以相比的。

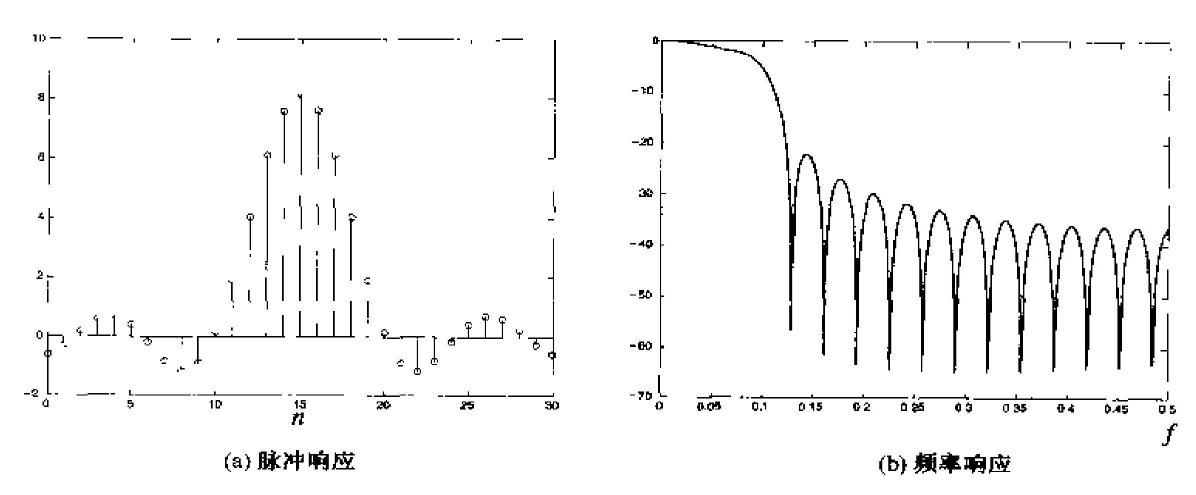


图 6.25　发送端截断的离散时间双二进制 FIR 滤波器的脉冲和频率响应

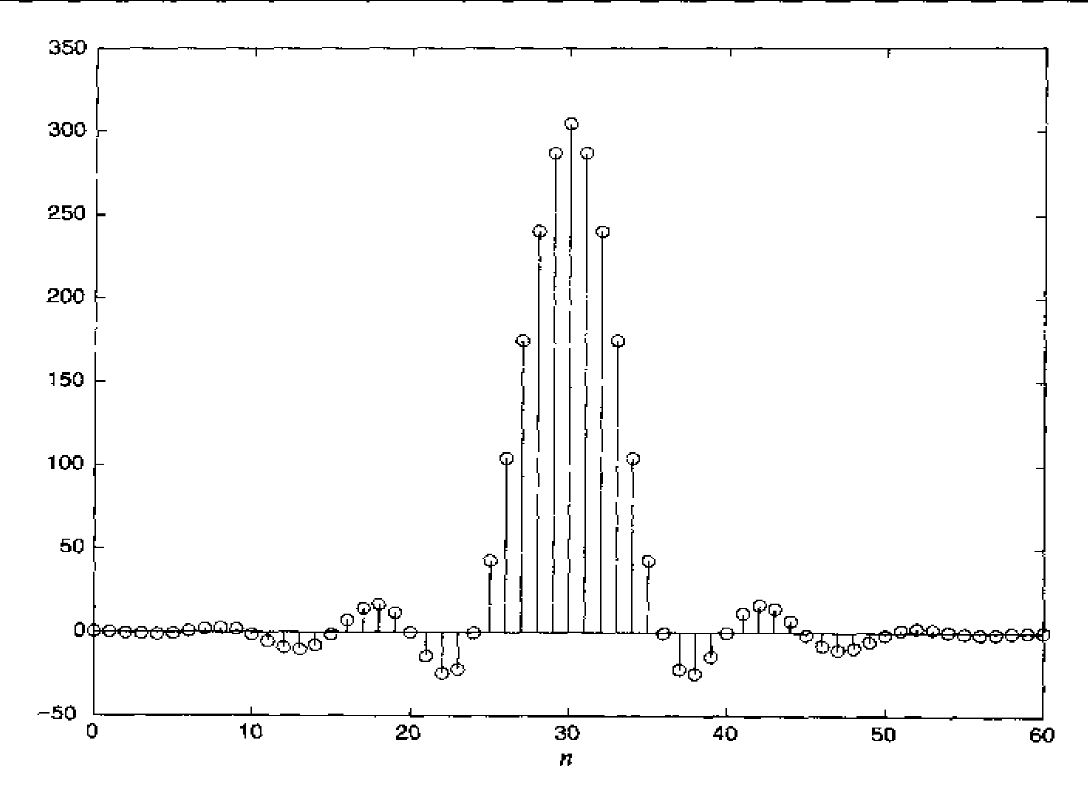


图 6.26　发送滤波器与接收端匹配滤波器级联后的脉冲响应

6.5.3　部分响应信号检测的预编码

对于二进制信号脉冲来说,有 $x(nT)=1, n=0,1$,而在其余 n 处,$x(nT)=0$,所以接收滤波器 $G_R(f)$的输出样本表示为:

$$\begin{aligned} y_k &= a_k + a_{k-1} + v_k \\ &= b_k + v_k \end{aligned} \tag{6.5.22}$$

其中,$\{a_k\}$是发送的幅度序列,$\{v_k\}$是加性高斯噪声样本序列和 $b_k = a_k + a_{k-1}$。先暂时忽略噪声并考虑具有等概率的二进制情况,这时 $a_k = \pm 1$。那么,b_k 就分别以概率$\frac{1}{4}$,$\frac{1}{2}$和$\frac{1}{4}$取 3 种可能的值,即 $b_k = -2, 0$ 和 2 中的一个。如果 a_{k-1}是从第$(k-1)$个信号区间检测的信号,那么它对第 k 个信号区间的接收信号 b_k 的影响就能通过相减而消除,由此能够检测出 a_k。这个过程可以对每个接收符号不断重复进行。

这个过程的主要问题是,由加性噪声引起的误差有传播的趋势。例如,如果 a_{k-1}是在差错中检测出的,那么它对 a_k 的影响没有消除;事实上由于不正确的相减反而增强了。因此,a_k 的检测也很可能有错误。

通过在发送端对数据**预编码**,而不是在接收端通过相减来消除这一可控制的 ISI,就能够防止误差传播。预编码是在调制之前对二进制数据序列实施的。由待发送的 0 和 1 的数据序列$\{D_k\}$产生一个称为**预编码序列**的新序列$\{p_k\}$。对双二进制信号来说,这个预编码序列定义为:

$$p_k = D_k \ominus p_{k-1}, \quad k=1,2,\cdots \tag{6.5.23}$$

其中，$\ominus$表示按模 2 减法[①]。这样，若 $p_k=0$，发送的信号幅度是 $a_k=-1$；若 $p_k=1$，则 $a_k=1$；也就是说

$$a_k=2p_k-1 \tag{6.5.24}$$

在接收滤波器输出端，无噪声样本由下式给出：

$$\begin{aligned} b_k &= a_k+a_{k-1} \\ &=(2p_k-1)+(2p_{k-1}-1) \\ &=2(p_k+p_{k-1}-1) \end{aligned} \tag{6.5.25}$$

这样

$$p_k+p_{k-1}=\frac{1}{2}b_k+1 \tag{6.5.26}$$

因为 $D_k=p_k\oplus p_{k-1}$，立刻利用下面的关系：

$$D_k=\frac{1}{2}b_k+1 \pmod 2 \tag{6.5.27}$$

就可以从$\{b_k\}$得到序列$\{D_k\}$。因此，若 $b_k=\pm 2$，则 $D_k=0$；若 $b_k=0$，则 $D_k=1$。表 6.1 给出了作为说明这个预编码和解码操作的一个例子。

表 6.1　具有双二进制脉冲的二进制信号

数据序列	D_k	–	1	1	1	0	1	0	0	1	0	0	0	1
预编码序列	p_k	0	1	0	1	1	0	0	0	1	1	1	1	0
发送序列	a_k	−1	1	−1	1	1	−1	−1	−1	1	1	1	1	−1
接收序列	b_k	–	0	0	0	2	0	−2	−2	0	2	2	2	0
解码序列	D_k	–	1	1	1	0	1	0	0	1	0	0	0	1

在加性噪声存在的情况下，接收滤波器的采样输出由式(6.5.22)给出。这时，$y_k=b_k+v_k$ 与两个阈值 +1 和 −1 做比较，按照下面的检测规则得到$\{D_k\}$序列：

$$D_k=\begin{cases}1, & |y_k|<1 \\ 0, & |y_k|\geqslant 1\end{cases} \tag{6.5.28}$$

这样，对数据预编码就能够完成在接收端的逐个符号检测，而不需要做前述的检测符号的相减。

利用双二进制脉冲从 2 电平 PAM 到多电平 PAM 的推广是很直接的。M 电平的传输序列$\{a_k\}$产生(无噪声)的接收序列

$$b_k=a_k+a_{k-1},\quad k=1,2,3,\cdots \tag{6.5.29}$$

它有 $2M-1$ 个可能的相等分隔开的幅度电平。由下面的关系：

$$a_k=2p_k-(M-1) \tag{6.5.30}$$

可以确定序列$\{a_k\}$的幅度电平，这里的$\{p_k\}$是预编码序列，它是从 M 个电平的数据序列$\{D_k\}$按下面的关系得到的：

$$p_k=D_k\ominus p_{k-1} \pmod M \tag{6.5.31}$$

---

① 虽然这个运算与按模 2 加法是相同的，但是对于双二进制信号，把预编码看成是按模 2 减法很方便。

这里的数据序列$\{D_k\}$的可能值是0,1,2,…,M。

在无噪声时,接收滤波器输出的样本可表示为:

$$\begin{aligned} b_k &= a_k + a_{k-1} \\ &= [2p_k - (M-1)] + [2p_{k-1} - (M-1)] \\ &= 2[p_k + p_{k-1} - (M-1)] \end{aligned} \tag{6.5.32}$$

所以

$$p_k + p_{k-1} = \frac{1}{2} b_k + (M-1) \tag{6.5.33}$$

因为 $D_k = p_k + p_{k-1} \pmod M$,于是发送数据$\{D_k\}$就利用下面的关系:

$$D_k = \frac{1}{2} b_k + (M-1) \quad (\mathrm{mod}\ M) \tag{6.5.34}$$

从接收序列$\{b_k\}$中恢复出来。在修正双二进制脉冲的情况下,接收滤波器 $G_R(f)$的输出端接收信号样本表示为:

$$y_k = a_k - a_{k-2} + v_k = b_k + v_k \tag{6.5.35}$$

对修正双二进制脉冲的预编码器,按下面的关系从数据序列$\{D_k\}$产生序列$\{p_k\}$:

$$p_k = D_k \oplus p_{k-2} \quad (\mathrm{mod}\ M) \tag{6.5.36}$$

由这些关系很容易证明,在无噪声情况下由$\{b_k\}$恢复数据序列$\{D_k\}$的检测规则是:

$$D_k = \frac{1}{2} b_k \quad (\mathrm{mod}\ M) \tag{6.5.37}$$

解说题

解说题 6.9

写一个 MATLAB 程序,该程序完成:取一个数据序列$\{D_k\}$,针对一个双二进制脉冲传输系统,对$\{D_k\}$预编码产生$\{p_k\}$,并将这个预编码序列映射到发送幅度电平$\{a_k\}$,然后从发送序列$\{a_k\}$形成接收无噪声序列$\{b_k\}$,再利用式(6.5.34)的关系恢复数据序列$\{D_k\}$。

题　解

本题的 MATLAB 脚本如下所示。利用这个程序,对 $M=2$,可以验证表 6.1 的结果。

m 文件

```
% MATLAB script for Illustrative Problem 6.9.
echo on
d=[1 1 1 0 1 0 0 1 0 0 0 1];
p(1)=0;
for i=1:length(d)
  p(i+1)=rem(p(i)+d(i),2);
  echo off ;
end
echo on ;
a=2.*p-1;
b(1)=0;
dd(1)=0;
for i=1:length(d)
  b(i+1)=a(i+1)+a(i);
  d_out(i+1)=rem(b(i+1)/2+1,2);
```

```
  echo off ;
end
echo on ;
d_out=d_out(2:length(d)+1);
```

6.6 线性均衡器

在实际情况下，用于降低 ISI 所采用的信道均衡器的最一般型式是线性 FIR 滤波器，它有可调节的系数 $\{c_k\}$，如图 6.27 所示。信道的频率响应特性是不知道的，而且是时变的，我们可测量出信道的特性并调节均衡器的参数，一旦调节好了，在数据传输过程中这些参数保持不变，这种均衡器称为**预置均衡器**(preset equalizer)。另外，**自适应均衡器**在数据传输过程中，在某一周期的基础上不断更新它们的参数，所以它们就有可能跟踪慢时变信道的响应。

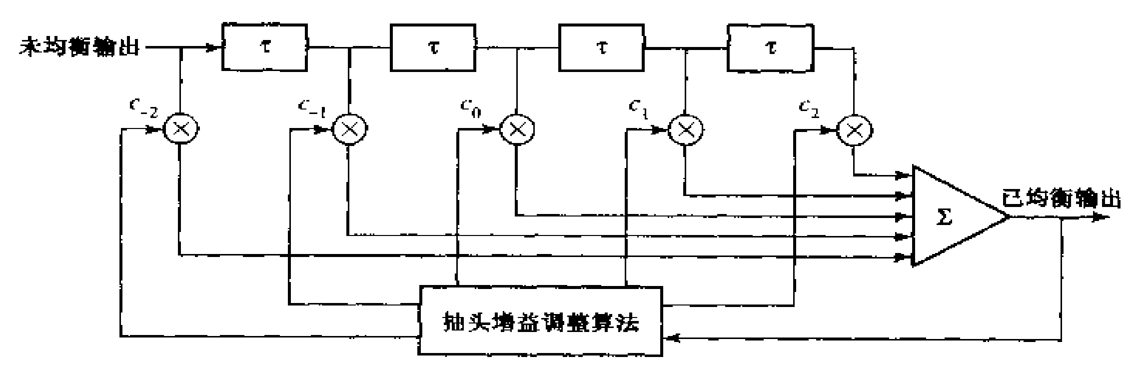


图 6.27　线性横向滤波器

首先，从频域的观点考虑一个线性均衡器的设计特性。图 6.28 给出了一个系统的方框图，它采用线性滤波器作为信道均衡器。解调器由频率响应为 $G_R(f)$ 的接收滤波器与频率响应为 $G_E(f)$ 的信道均衡器的级联组成。如同前一节所述，接收滤波器的频率响应 $G_R(f)$ 是与发送滤波器的频率响应 $G_T(f)$ 相匹配的，即 $G_R(f)=G_T^*(f)$，而乘积 $G_R(f)G_T(f)$ 通常或者设计成在采样瞬时具有零 ISI 的特性(例如，当 $G_R(f)G_T(f)=X_{rc}(f)$ 时就属于这种情况)，或者对部分响应信号具有可控的 ISI 特性。

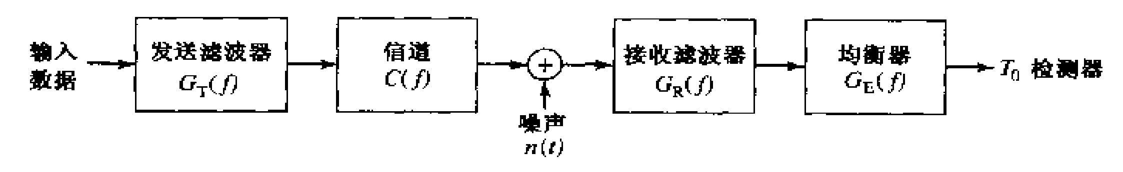


图 6.28　具有均衡器的系统方框图

对于图 6.28 所示的系统，信道频率响应不是理想的，因此对于零 ISI 的条件就是：

$$G_T(f)C(f)G_R(f)G_E(f)=X_{rc}(f) \tag{6.6.1}$$

其中，$X_{rc}(f)$ 是期望的升余弦频谱特性。因为按设计 $G_T(f)G_R(f)=X_{rc}(f)$，所以补偿信道失真的均衡器的频率响应为：

$$G_E(f)=\frac{1}{C(f)}=\frac{1}{|C(f)|}e^{-j\theta_c(f)} \tag{6.6.2}$$

由此,均衡器的幅度响应是$|G_E(f)|=1/|C(f)|$,相位响应是$\theta_E(f)=-\theta_c(f)$。在这种情况下就说该均衡器是对信道响应的**逆信道滤波器**。

注意,这个逆信道滤波器完全消除了由信道引起的ISI。因为这种均衡强迫使ISI在采样瞬时$t=kT,k=0,1,\cdots$为零,所以称类均衡器为**强迫归零均衡器**(zero-forcing equalizer)。因此,检测器的输入就是:

$$z_k=a_k+\eta_k,\quad k=0,1,\cdots \tag{6.6.3}$$

其中,η_k代表加性噪声,a_k是期望的符号。

实际上,由信道失真引起的ISI通常都将它局限在所期望符号两边的有限个符号数上,所以由式(6.4.7)给出的求和式中构成ISI的项数是有限的。这样,实际的信道均衡器是按有限脉冲响应(FIR)滤波器或横向滤波器实现的,它具有可供调节的抽头系数$\{c_n\}$,如图6.27所示。相邻抽头之间的延时τ可以选成和T(即符号区间)一样大。在这种情况下,均衡器的输入是由式(6.4.6)给出的采样序列。然而,应该注意到,当符号率$1/T<2W$时,在接收信号中超过折叠频率$1/T$的那些频率被混叠到低于$1/T$的频率中,这时这个均衡器对这个已混叠的信道失真信号给予补偿。

另一方面,当相邻抽头之间的延时τ选成$\frac{1}{\tau}\geqslant 2W>\frac{1}{T}$时不发生混叠,因此该逆信道均衡器对真正的信道失真给予补偿。因为$\tau<T$,就说该信道均衡器具有fractionally spaced taps(部分符号间隔抽头),而称之为fractionally spaced equalizer(部分符号间隔均衡器)。实际上τ往往选为$\tau=T/2$。要注意,这时在滤波器$G_E(f)$的输入端的采样率是$2/T$。

该FIR均衡器的冲激响应是:

$$g_E(t)=\sum_{n=-K}^{K}c_n\delta(t-n\tau) \tag{6.6.4}$$

相应的频率响应是:

$$G_E(f)=\sum_{n=-K}^{K}c_n\mathrm{e}^{-\mathrm{j}2\pi fn\tau} \tag{6.6.5}$$

其中,$\{c_n\}$是$2K+1$个均衡器系数,而K选得足够大,以使均衡器跨过ISI的长度,也即$2K+1\geqslant L$,L是由ISI跨过的信号样本个数。因为$X(f)=G_T(f)C(f)G_R(f)$,而$x(t)$就是对应于$X(f)$的信号脉冲,所以已均衡的输出信号脉冲是:

$$q(t)=\sum_{n=-K}^{K}c_nx(t-n\tau) \tag{6.6.6}$$

现在应该对在$t=mT$取得的$q(t)$样本应用归零条件。这些样本是:

$$q(mT)=\sum_{n=-K}^{K}c_nx(mT-n\tau),\quad m=0,\pm1,\cdots,\pm K \tag{6.6.7}$$

因为有$2K+1$个均衡器系数,所以仅能控制$2K+1$个$q(t)$的采样值。具体一些就是可以强迫使这些条件为:

$$q(mT)=\sum_{n=-K}^{K}c_nx(mT-n\tau)=\begin{cases}1, & m=0\\0, & m=\pm1,\pm2,\cdots,\pm K\end{cases} \tag{6.6.8}$$

它可以表示成矩阵形式：$\boldsymbol{Xc} = \boldsymbol{q}$，其中 $\boldsymbol{X}$ 是 $(2K+1)\times(2K+1)$ 的矩阵，其元素为 $x(mT-n\tau)$，$\boldsymbol{c}$ 是 $(2K+1)$ 的系数向量，$\boldsymbol{q}$ 是 $(2K+1)$ 个列向量，其中只有一个非零元素。这样就得到一组具有强迫归零均衡器系数的 $(2K+1)$ 个线性方程组。

应该强调的是，这个 FIR 强迫归零均衡器并没有完全消除 ISI，因为它是有限长的。然而，随着 K 的增加，残留的 ISI 可以减小，在 $K\to\infty$ 的极限情况下，ISI 完全被消除。

解说题

解说题 6.10

考虑一个信道失真了的脉冲 $x(t)$，在均衡器的输入端由下式表示：

$$x(t) = \frac{1}{1+(2t/T)^2}$$

其中，$1/T$ 是符号率。该脉冲在 $2/T$ 采样率被采样，而且用一个强迫归零均衡器给予均衡。求具有 5 个抽头的强迫归零均衡器的系数。

题　解

根据式(6.6.8)，该强迫归零均衡器必须满足下式：

$$q(mT) = \sum_{n=-2}^{2} c_n x\left(mT - \frac{nT}{2}\right) = \begin{cases} 1, & m = 0 \\ 0, & m = \pm 1,\ \pm 2 \end{cases}$$

矩阵 $\boldsymbol{X}$ 有元素为 $x(mT - nT/2)$，由下式给出：

$$\boldsymbol{X} = \begin{bmatrix} \frac{1}{5} & \frac{1}{10} & \frac{1}{17} & \frac{1}{26} & \frac{1}{37} \\ 1 & \frac{1}{2} & \frac{1}{5} & \frac{1}{10} & \frac{1}{17} \\ \frac{1}{5} & \frac{1}{2} & 1 & \frac{1}{2} & \frac{1}{5} \\ \frac{1}{17} & \frac{1}{10} & \frac{1}{5} & \frac{1}{2} & 1 \\ \frac{1}{37} & \frac{1}{26} & \frac{1}{17} & \frac{1}{10} & \frac{1}{5} \end{bmatrix} \tag{6.6.9}$$

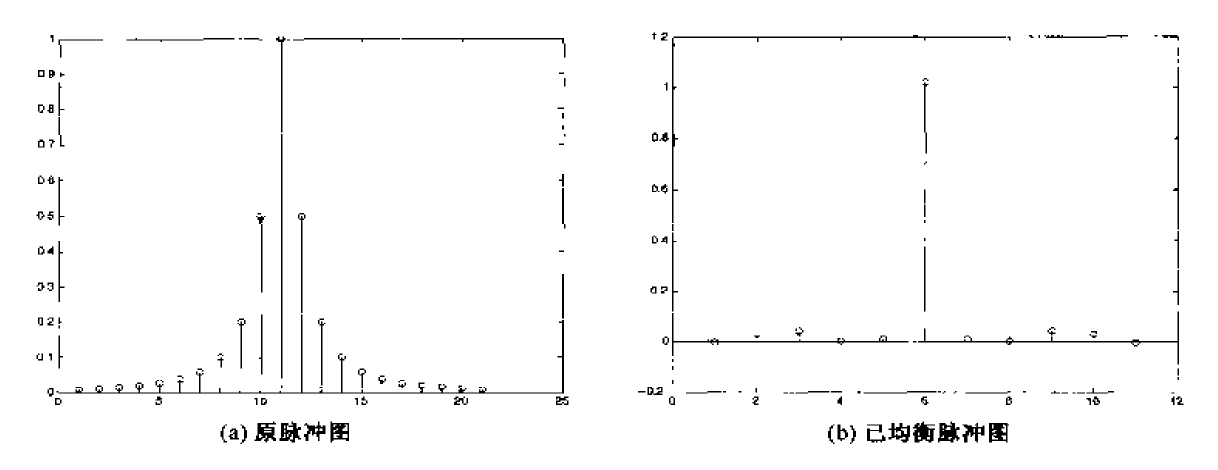
系数向量 $\boldsymbol{c}$ 和向量 $\boldsymbol{q}$ 给出如下：

$$\boldsymbol{c} = \begin{bmatrix} c_{-2} \\ c_{-1} \\ c_0 \\ c_1 \\ c_2 \end{bmatrix} \qquad \boldsymbol{q} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \tag{6.6.10}$$

通过求矩阵 $\boldsymbol{X}$ 的逆，解出这个线性方程 $\boldsymbol{Xc} = \boldsymbol{q}$，求得：

$$\boldsymbol{c}_{\text{opt}} = \boldsymbol{X}^{-1}\boldsymbol{q} = \begin{bmatrix} -2.2 \\ 4.9 \\ -3 \\ 4.9 \\ -2.2 \end{bmatrix} \tag{6.6.11}$$

图 6.29 给出了原脉冲 $x(t)$ 和已均衡脉冲。注意，在已均衡脉冲中仍有小的残余 ISI 量。



(a) 原脉冲图　　(b) 已均衡脉冲图

图 6.29　解说题 6.10 中的脉冲图

计算本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 6.10.
echo on
T=1;
Fs=2/T;
Ts=1/Fs;
c_opt=[-2.2 4.9 -3 4.9 -2.2];
t=-5*T:T/2:5*T;
x=1./(1+((2/T)*t).^2);                    % sampled pulse
equalized_x=filter(c_opt,1,[x 0 0]);      % since there will be a delay of two samples at the output
% to take care of the delay
equalized_x=equalized_x(3:length(equalized_x));
% Now, let us downsample the equalizer output.
for i=1:2:length(equalized_x),
  downsampled_equalizer_output((i+1)/2)=equalized_x(i);
  echo off;
end;
echo on ;
% Plotting commands follow.
```

强迫归零均衡器的一个缺点是没有考虑噪声的存在,因此使用了它以后可能会形成显著的噪声增强。这一点很容易看出来。注意,在 $C(f)$比较小的频率范围中,信道均衡器$G_E(f) = 1/C(f)$就在这段频率范围内通过大的增益来进行补偿,其后果是在这段频率范围内的噪声也大为增强。一种替代方法是,放松零 ISI 条件并选择信道均衡器特性,使得剩余的 ISI 和均衡器输出端的加性噪声的组合功率最小。基于最小均方误差(MMSE)准则优化的信道均衡器能实现所期望的目标。

为了详细阐述,先考虑 FIR 均衡器受噪声污损的输出为:

$$z(t) = \sum_{n=-K}^{K} c_n y(t - n\tau) \tag{6.6.12}$$

其中，$y(t)$是均衡器的输入，由式(6.4.3)给出。均衡器输出在 $t=mT$ 时刻采样，可得：

$$z(mT)=\sum_{n=-K}^{K}c_n y(mT-n\tau) \tag{6.6.13}$$

在 $t=mT$ 时，均衡器输出的期望响应就是发送符号 a_m。a_m 和 $z(mT)$之间的差定义为误差，那么实际输出样本 $z(mT)$和期望值 a_m 之间的均方误差(MSE)就为[①]：

$$\begin{aligned}\text{MSE} &= E\,|z(mT)-a_m|^2\\ &= E\left[\left|\sum_{n=-K}^{K}c_n y(mT-n\tau)-a_m\right|^2\right]\\ &= \sum_{n=-K}^{K}\sum_{k=-K}^{K}c_n c_k R_y(n-k)-2\sum_{k=-K}^{K}c_k R_{ay}(k)+E(|a_m|^2)\end{aligned} \tag{6.6.14}$$

其中的相关定义为：

$$\begin{aligned}R_y(n-k)&=E[y^*(mT-n\tau)y(mT-k\tau)]\\ R_{ay}(k)&=E[y(mT-k\tau)a_m^*]\end{aligned} \tag{6.6.15}$$

期望是对随机信息序列$\{a_m\}$和加性噪声取的。

将式(6.6.14)对均衡器系数$\{c_n\}$微分，可得最小 MSE 的解，于是对最小 MSE 得出的必要条件是：

$$\sum_{n=-K}^{K}c_n R_y(n-k)=R_{ay}(k),k=0,\pm1,\pm2,\cdots,\pm K \tag{6.6.16}$$

这是均衡器系数的 $2K+1$ 个线性方程组。和前面讨论过的归零解不同，这些方程都与噪声的统计性质(自相关)以及通过自相关函数 $R_y(n)$的 ISI 统计性质有关。

在实际情况下，自相关矩阵 $\boldsymbol{R}_y(n)$和互相关向量 $\boldsymbol{R}_{ay}(n)$事先都是不知道的。然而，这些相关序列可通过在信道上发送某一测试信号并采用时间平均估值而估计出来：

$$\begin{aligned}\hat{R}_y(n)&=\frac{1}{K}\sum_{k=1}^{K}y^*(kT-n\tau)y(kT)\\ \hat{R}_{ay}(n)&=\frac{1}{K}\sum_{k=1}^{K}y(kT-n\tau)a_k^*\end{aligned} \tag{6.6.17}$$

替代了为解由式(6.6.16)给出的均衡器系数而要求的集合平均。

解说题

解说题 6.11

再次考虑在解说题 6.10 中的同一个信道失真信号 $x(t)$，但现在要基于最小 MSE 准则设计 5 个抽头的均衡器。信息符号具有零均值和单位方差，并且互不相关，即

$$\begin{aligned}&E(a_n)=0\\ &E(a_n a_m)=0,\quad n\neq m\\ &E(|a_n|^2)=1\end{aligned}$$

加性噪声 $v(t)$有零均值和自相关函数：

① 在导出过程中，允许信号 $z(t)$和 $y(t)$是复值，也允许数据序列是复值。在第 7 章中讨论载波调制信号时也是如此。

$$\varphi_w(\tau)=\frac{N_0}{2}\delta(\tau)$$

题 解

用 $K=2$ 和 $\tau=T/2$ 解式(6.6.16),得出均衡器的抽头系数。元素为 $R_y(n-k)$的矩阵是:

$$\boldsymbol{R}_y=\boldsymbol{X}^{t}\boldsymbol{X}+\frac{N_0}{2}\boldsymbol{I}$$

其中的 $\boldsymbol{X}$ 由式(6.6.9)给出,$\boldsymbol{I}$ 是恒等矩阵。元素为 $R_{ay}(k)$的向量为:

$$\boldsymbol{R}_{ay}=\begin{bmatrix}\frac{1}{5}\\ \frac{1}{2}\\ 1\\ \frac{1}{2}\\ \frac{1}{5}\end{bmatrix}$$

对式(6.6.16)解出的均衡器系数如下:

$$\boldsymbol{c}_{\text{opt}}=\begin{bmatrix}0.0956\\ -0.7347\\ 1.6761\\ -0.7347\\ 0.0956\end{bmatrix}$$

图 6.30 给出了已均衡脉冲的图。

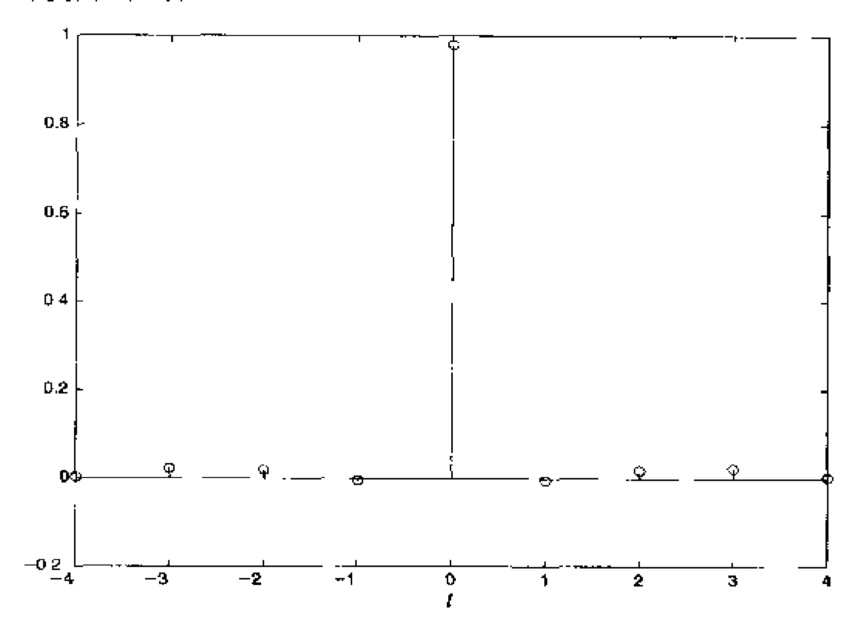


图 6.30 解说题 6.11 中的已均衡脉冲的图

计算本题的 MATLAB 脚本如下所示。

```
% MATLAB script for Illustrative Problem 6.11.
echo on
T=1;
for n=-2:2,
  for k=-2:2,
    temp=0;
    for i=-2:2, temp=temp+(1/(1+(n-i)^2))*(1/(1+(k-i)^2)); end;
    X(k+3,n+3)=temp;
    echo off ;
  end;
end;
echo on;
N0=0.01;                              % assuming that N0=0.01
Ry=X+(N0/2)*eye(5);
Riy=[1/5 1/2 1 1/2 1/5].';
c_opt=inv(Ry)*Riy;                    % optimal tap coefficients
% find the equalized pulse...
t=-3:1/2:3;
x=1./(1+(2*t/T).^2);                  % sampled pulse
equalized_pulse=conv(x,c_opt);
% Decimate the pulse to get the samples at the symbol rate.
decimated_equalized_pulse=equalized_pulse(1:2:length(equalized_pulse));
% Plotting command follows.
```

6.6.1 自适应线性均衡器

我们已经证明一个线性均衡器的抽头系数可以通过解一组线性方程来确定。在归零最佳准则中，这组线性方程由式(6.6.8)给出；另一方面，若是基于最小 MSE 的最佳准则，那么最佳均衡器系数由解式(6.6.16)给出的这组线性方程确定。

在两种情况下，可以将这组线性方程表示成如下的一般矩阵形式：

$$\boldsymbol{B}\boldsymbol{c} = \boldsymbol{d} \tag{6.6.18}$$

其中，$\boldsymbol{B}$ 是$(2K+1)\times(2K+1)$的矩阵，$\boldsymbol{c}$ 是代表$(2K+1)$个均衡器系数的列向量，而 $\boldsymbol{d}$ 是$(2K+1)$维列向量。式(6.6.18)的解为：

$$\boldsymbol{c}_{\text{opt}} = \boldsymbol{B}^{-1}\boldsymbol{d} \tag{6.6.19}$$

在均衡器的实际实现中，通常式(6.6.18)对最佳系数向量的解是用迭代过程来得到的，这可以避免明确地计算矩阵 $\boldsymbol{B}$ 的逆。最简单的迭代过程是最陡下降法，其中任意选取系数向量 $\boldsymbol{c}$，比如 $\boldsymbol{c}_0$ 作为起始值。系数向量 $\boldsymbol{c}_0$ 的初始选择对应于正在优化的判据函数上的某个点，例如在 MSE 准则的情况下，初始推测值 $\boldsymbol{c}_0$ 对应于$(2K+1)$维系数空间中二次 MSE 曲面上的某个点。定义为梯度向量的 $\boldsymbol{g}_0$ 是 MSE 对 $2K+1$ 个滤波器系数的导数，然后在判据曲面上在这个点将 $\boldsymbol{g}_0$ 计算出来，而且每个抽头系数都朝着与对应的梯度分量相反的方向改变，在第 j 个抽头系数上的变化正比于第 j 个梯度分量的大小。

例如，梯度向量表示成 $\boldsymbol{g}_k$，对于 MSE 准则，取 MSE 对 $2K+1$ 个系数中的每一个的导数后，求得：

$$\boldsymbol{g}_k = \boldsymbol{B}\boldsymbol{c}_k - \boldsymbol{d}, \quad k=0,1,2,\cdots \tag{6.6.20}$$

然后，系数向量 $\boldsymbol{c}_k$ 按下面的关系更新：

$$\boldsymbol{c}_{k+1}=\boldsymbol{c}_k-\Delta\boldsymbol{g}_k \tag{6.6.21}$$

其中,Δ是迭代过程中的**步长参数**(step-size parameter)。为了确保这个迭代过程收敛,Δ选一个小的正数。在这种情况下,梯度向量 $\boldsymbol{g}_k$ 收敛到零,即当 $k\to\infty$, $\boldsymbol{g}_k\to 0$,而系数向量 $\boldsymbol{c}_k\to\boldsymbol{c}_{opt}$ 时,基于二维优化的情况如图6.31所示。一般来说,均衡器抽头系数收敛到 $\boldsymbol{c}_{opt}$ 是不能用最陡下降法经有限次迭代达到的。然而,在经过几百次迭代之后,就能按要求接近于最优解 $\boldsymbol{c}_{opt}$。在采用信道均衡器的数字通信系统中,每次迭代对应于发送一个符号的时间间隔,所以为实现收敛到 $\boldsymbol{c}_{opt}$ 的几百次迭代也就相应于几分之一秒。

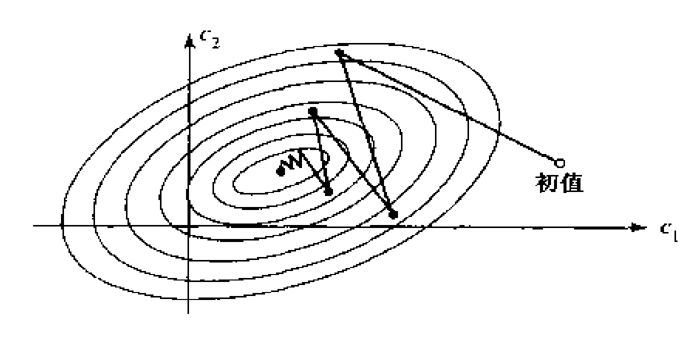


图6.31 梯度算法的收敛特性举例

对于频率响应特性随时间变化的信道来说,需要有自适应信道均衡。在这种情况下,ISI也随时间变化。信道均衡器必须跟踪在信道响应上的这样的时间波动,并将它的系数自适应,以降低ISI。在上面讨论的范畴内,由于矩阵 $\boldsymbol{B}$ 随时间变化,以及在MSE准则下向量 $\boldsymbol{d}$ 随时间变化,因此最佳系数向量 $\boldsymbol{c}_{opt}$ 也是随时间变化的。在这些条件下,可以将上面讨论的迭代法修改为利用梯度分量估计值。因此,调节均衡器抽头系数的算法可以表示为:

$$\hat{\boldsymbol{c}}_{k+1}=\hat{\boldsymbol{c}}_k-\Delta\hat{\boldsymbol{g}}_k \tag{6.6.22}$$

其中,$\hat{\boldsymbol{g}}_k$ 代表梯度向量 $\boldsymbol{g}_k$ 的估值,$\hat{\boldsymbol{c}}_k$ 代表抽头系数向量的估值。

在MSE准则的情况下,由式(6.6.20)给出的梯度向量 $\hat{\boldsymbol{g}}_k$ 的估值,也可以表示为:

$$\hat{\boldsymbol{g}}_k=-E(\mathrm{e}_k\boldsymbol{y}_k^*)$$

梯度向量在第 k 次迭代的估值 $\hat{\boldsymbol{g}}_k$ 可计算为:

$$\hat{\boldsymbol{g}}_k=-\mathrm{e}_k\boldsymbol{y}_k^* \tag{6.6.23}$$

其中,e_k 表示在第 k 个时间瞬时均衡器的期望输出与实际输出 $z(kT)$ 之差,$\boldsymbol{y}_k$ 代表在瞬时 k 包含均衡器中 $2K+1$ 接收信号值的列向量。误差信号 e_k 表示为:

$$\mathrm{e}_k=a_k-z_k \tag{6.6.24}$$

其中,$z_k=z(kT)$ 是由式(6.6.13)给出的均衡器输出,a_k 是期望的符号。这样,将式(6.6.23)代入式(6.6.22),得到优化抽头系数(基于MSE准则)的自适应算法为:

$$\hat{\boldsymbol{c}}_{k+1}=\hat{\boldsymbol{c}}_k+\Delta\mathrm{e}_k\boldsymbol{y}_k^* \tag{6.6.25}$$

因为在式(6.6.25)中用的是梯度向量的估值,所以这个算法称为**随机梯度算法**,也称为**LMS算法**。

将抽头系数按式(6.6.25)自适应的自适应均衡器的方框如图6.32所示。值得注意的是,期望的输出 a_k 和来自均衡器的实际输出 z_k 之差用来构成误差信号 e_k,这个误差被步长参数Δ加权,加权后的误差信号 $\Delta\mathrm{e}_k$ 乘以 $2K+1$ 节抽头的接收信号值 $\{y(kT-n\tau)\}$。将 $2K+1$ 节抽

头的乘积 $\Delta e_k y^*(kT - n\tau)$ 按照式(6.6.25)加到这些抽头系数以前的值上，以得到更新了的抽头系数。每接收一个新的信号样本，这个计算就重复一次，因此均衡器的系数以符号率更新。

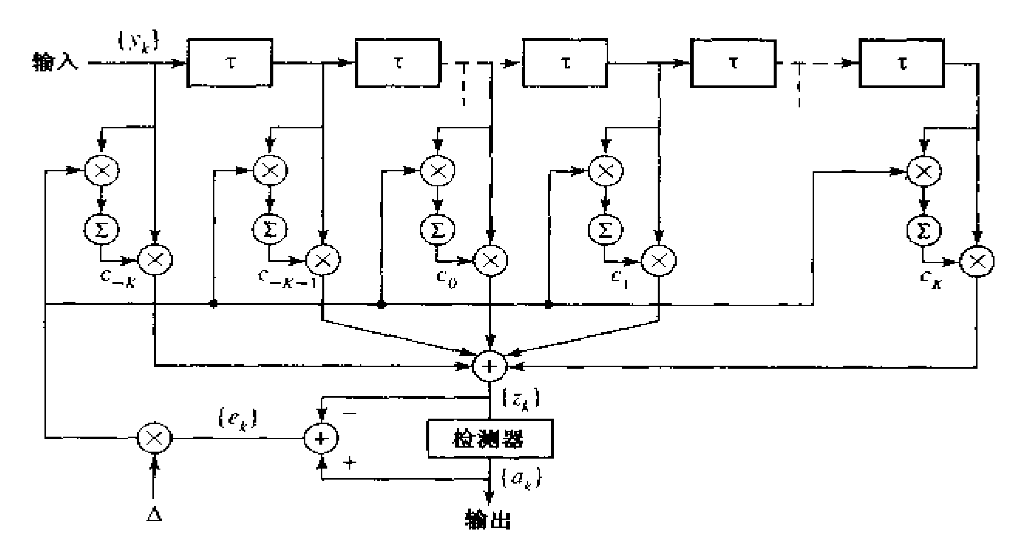


图 6.32　基于 MSE 准则的线性自适应均衡器

最初用一个已知伪随机序列 $\{a_m\}$ 在信道上训练这个自适应均衡器。在解调器端，均衡器用这个已知序列去调整它的系数，一旦初始调节完成，自适应均衡器就从一个**训练模式**(training mode)切换到**直接判决模式**(decision-directed mode)。在这种情况下，在检测器输出端的这些判决足够可靠，因此可以通过计算检测器输出和均衡器输出之间的差来形成误差信号，即：

$$e_k = \hat{a}_k - z_k \tag{6.6.26}$$

其中，$\hat{a}_k$ 是检测器的输出。一般在检测器输出的判决差错很少发生，因而这样的误差对由式(6.6.25)给出的跟踪算法的性能几乎没有多少影响。

为了确保收敛和在慢变化信道中的好的跟踪能力，选择步长参数的一种经验公式是：

$$\Delta = \frac{1}{5(2K+1)P_R} \tag{6.6.27}$$

其中，P_R 代表接收到的信号加噪声的功率，它可以从接收信号中估计出来。

解说题

解说题 6.12

基于由式(6.6.25)给出的 LMS 算法实现一个自适应均衡器。均衡选取的信道抽头数是 $2K+1=11$。接收信号加噪声的功率 P_R 归一化到 1。信道特性由向量 $\boldsymbol{x}$ 给出：

$\boldsymbol{x} = (0.05, -0.063, 0.088, -0.126, -0.25, 0.9047, 0.250, 0.126, 0.038, 0.088)$

题　解

式(6.6.25)随机梯度算法的收敛特性如图 6.33 所示。这些曲线是用 11 个抽头系数的自适应均衡器的计算机仿真得出的，曲线代表几次实现后平均的均方误差。由此可见，当 Δ 减小时，收敛稍许变慢，但可达到更低的 MSE，这表明估值的系数更接近 $\boldsymbol{c}_{opt}$。

本例的 MATLAB 脚本如下所示。

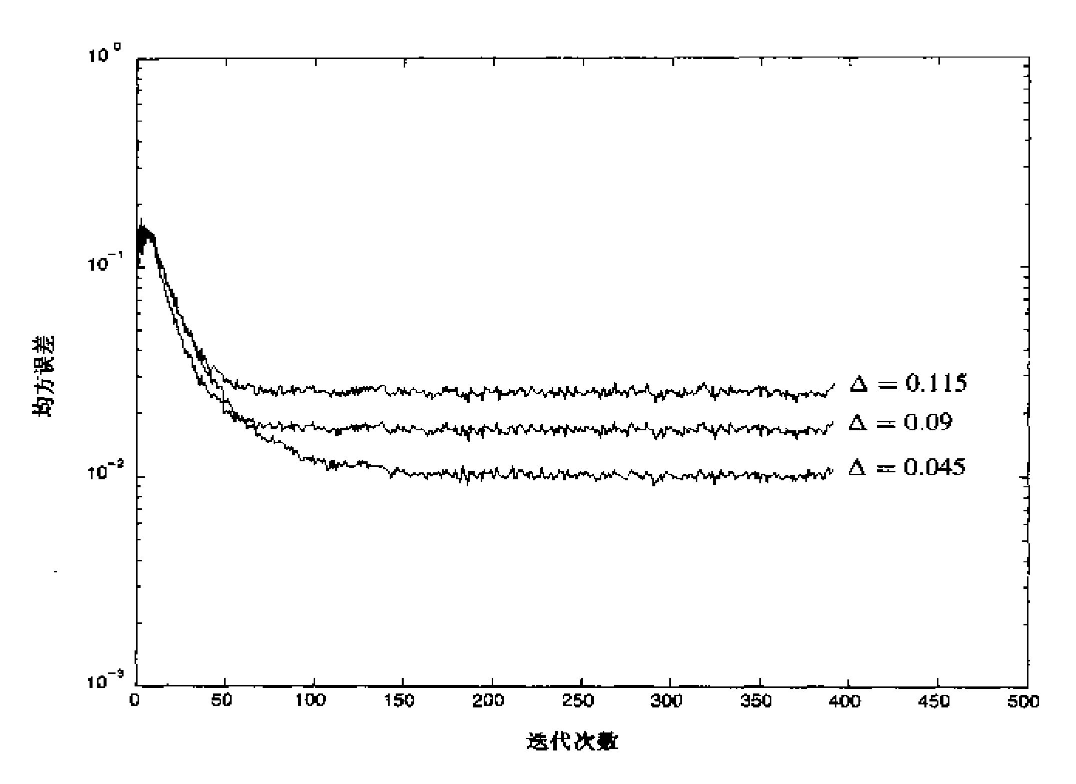


图 6.33 不同步长下 LMS 算法的最初收敛特性

m 文件

```
% MATLAB script for Illustrative Problem 6.12.
echo on
N=500;                                  % length of the information sequence
K=5;
actual_isi=[0.05 -0.063 0.088 -0.126 -0.25 0.9047 0.25 0 0.126 0.038 0.088];
sigma=0.01;
delta=0.115;
Num_of_realizations=1000;
mse_av=zeros(1,N-2*K);
for j=1:Num_of_realizations,            % Compute the average over a number of realizations.
  % the information sequence
  for i=1:N,
    if (rand<0.5),
      info(i)=-1;
    else
      info(i)=1;
    end;
    echo off ;
  end;
  if (j==1) ; echo on ; end
  % the channel output
  y=filter(actual_isi,1,info);
  for i=1:2:N, [noise(i) noise(i+1)]=gngauss(sigma); end;
  y=y+noise;
  % Now the equalization part follows.
  estimated_c=[0 0 0 0 0 1 0 0 0 0 0];  % initial estimate of ISI
  for k=1:N-2*K,
    y_k=y(k:k+2*K);
    z_k=estimated_c*y_k.' ;
```

```
      e_k=info(k)-z_k;
      estimated_c=estimated_c+delta*e_k*y_k;
      mse(k)=e_k^2;
      echo off ;
    end;
    if (j==1) ; echo on ; end
    mse_av=mse_av+mse;
    echo off ;
end;
echo on ;
mse_av=mse_av/Num_of_realizations;          % mean-squared error versus iterations
% Plotting commands follow.
```

我们只比较详细地讨论了基于 MSE 准则优化的自适应均衡器的工作过程，基于归零方法的自适应均衡器的工作也是类似的。主要的差别在于每次迭代中梯度向量 g_k 的估值不同。一种自适应强迫归零均衡器的方框图如图 6.34 所示。

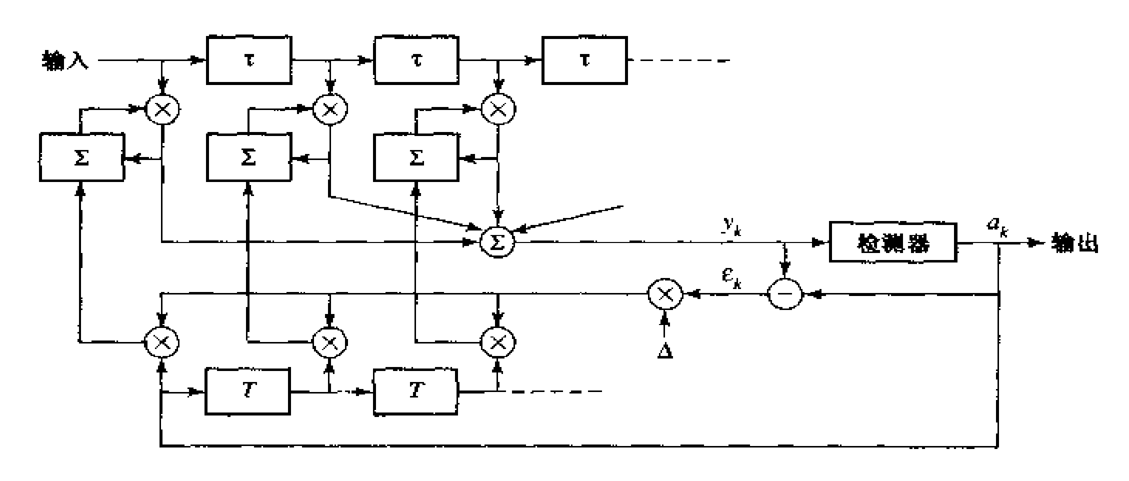


图 6.34　自适应强迫归零均衡器

6.7　非线性均衡器

前面讨论的线性均衡器在诸如有线电话之类的信道上是很有效的，这种情况下 ISI 不是很严重。ISI 的严重性直接与信道的谱特性有关，而不一定与 ISI 的时间跨度有关。例如，考虑如图 6.35 所示的两条信道所形成的 ISI。在信道 A 上 ISI 的时间性跨度在期望信号分量的两边是 5 个符号间隔，期望信号分量值是 0.72；而在信道 B 上 ISI 的时间跨度在期望信号的两边是 1 个符号间隔，其值是 0.815。这两条信道总的响应能量都归一化到 1。

尽管 ISI 的跨度较短，但是信道 B 却产生更为严重的 ISI。这一点从两个信道的频率响应特性上看很明显，如图 6.36 所示。由图中可以看到，信道 B 在 $f=1/(2T)$ 有一个谱为零(在频带 $|f|\leqslant W$ 内某些频率处频率响应 $C(f)=0$)，而这种情况在信道 A 中并不会发生。因此，一个线性均衡器将会在它的频率响应中引入一个大的增益，以补偿这个信道的零点，由此在信道 B 中的噪声也就会比信道 A 大大增强。这意味着信道 B 的线性均衡器的性能比信道 A 的差得多。一般来说，线性均衡器的基本局限就在于对频谱有零域的信道无能为力。这样的信道往

往在无线通信中会遇到,如低于 30 MHz 频率的电离层传输、移动无线信道以及用于蜂窝无线通信中的那些信道。

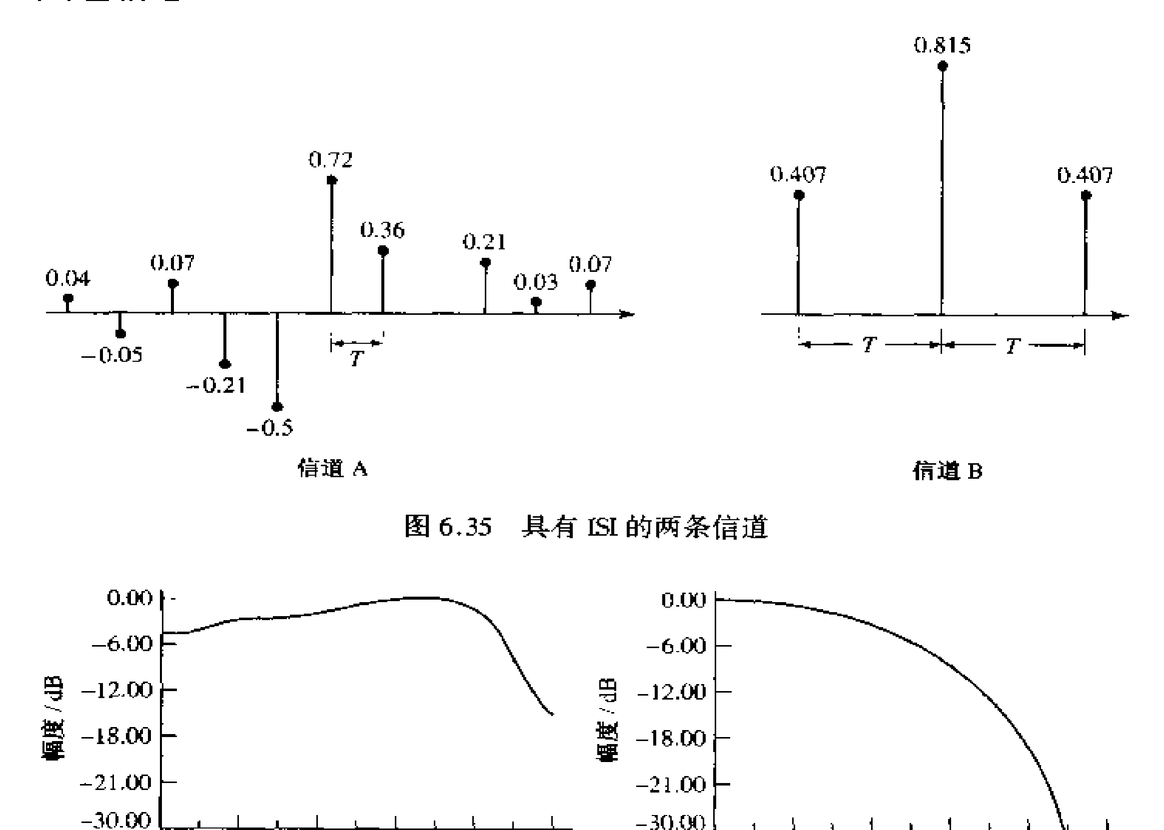


图 6.35　具有 ISI 的两条信道

图 6.36　由图 6.35 得到的幅度谱

反馈判决均衡器(Decision-Feedback Equalizer,简称 DFE)是一种非线性均衡器,它利用先前的判决来消除由前面检测出的符号在当前待检测符号上产生的 ISI。图 6.37 给出了一个简单的 DFE 方框图。DFE 由两个滤波器组成。第一个滤波器称为**前馈滤波器**,一般它是具有可调节抽头系数的部分符号间隔的 FIR 滤波器。这个滤波器在形式上与早先讨论的线性均衡器是相同的,它的输入是接收过滤的信号 $y(t)$ 以某个符号率的倍数采样的采样值,比如 $2/T$ 的采样率。第二个滤波器是**反馈滤波器**,它是作为以符号间隔抽头的可调节系数的 FIR 滤波器实现的。它的输入是一组先前已检测出的符号。将反馈滤波器的输出从前馈滤波器的输出中减去,即可形成检测器的输入。因此有:

$$z_m = \sum_{n=1}^{N_1} c_n y(mT - n\tau) - \sum_{n=1}^{N_2} b_n \tilde{a}_{m-n}$$

其中,$\{c_n\}$ 和 $\{b_n\}$ 分别是前馈和反馈滤波器的可调节系数;$\tilde{a}_{m-n}, n = 1,2,\cdots$;$N_2$ 是前面检测出的符号;N_1 是前馈滤波器的长度;N_2 是反馈滤波器的长度。根据输入 z_m,检测器判断哪一个可能的发送符号与输入信号 a_m 的距离最接近,据此做出判决并输出 $\tilde{a}_m$。使 DFE 非线性的是检测器的非线性特性,它为反馈滤波器提供了输入。

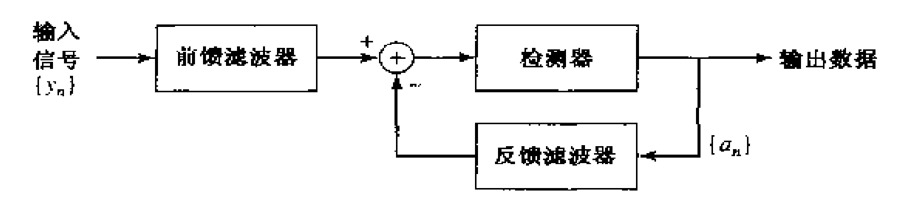


图 6.37 DFE 方框图

前馈和反馈滤波器的抽头系数的选取是为了优化某个期望的性能指标。为了使数学上简单，通常都采用 MSE 准则，而随机梯度算法一般用于实现自适应 DFE。图 6.38 给出了一个自适应 DFE 的方框图，它的抽头系数是用 LMS 随机梯度算法调节的。

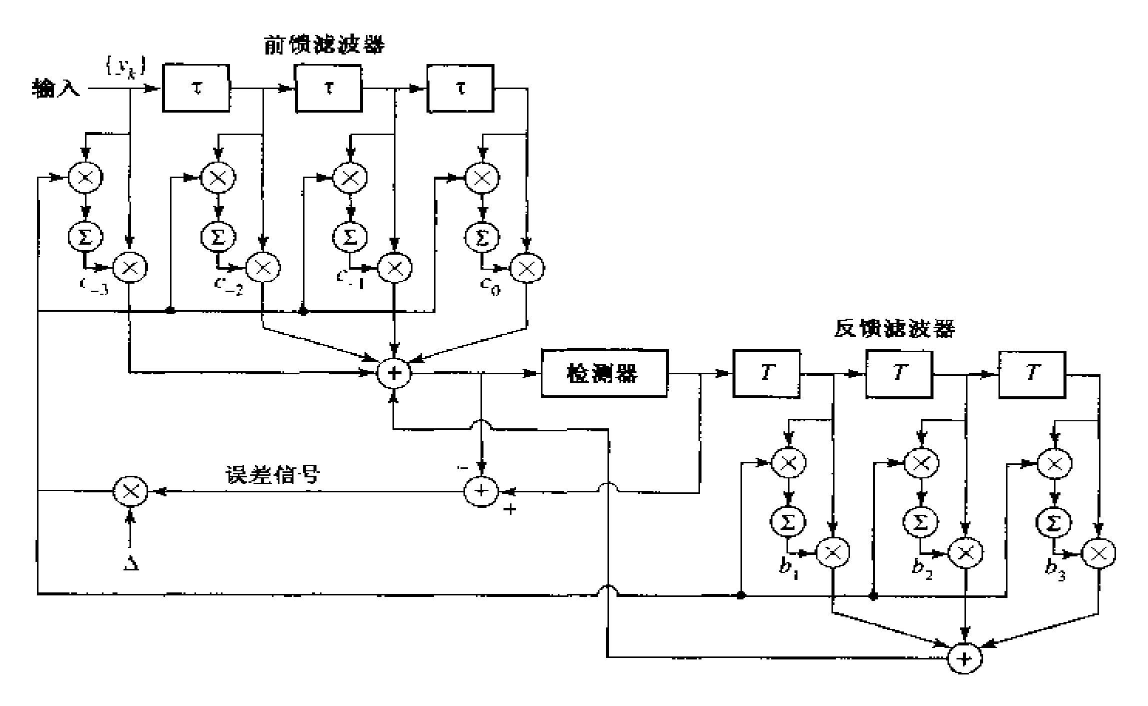


图 6.38 自适应 DFE 的方框图

应该提及，馈给反馈滤波器的来自检测器的判决误差对 DFE 的性能只有小的影响。一般在误码率低于 10^{-2}的情况下可能有 1 ~ 2 dB 的性能损失，但是判决误差对于反馈滤波器不会有大的影响。

尽管 DFE 要比线性均衡器更好一些，但是在由式(6.4.6)给出的接收信号样本$\{y_k\}$的信息序列$\{a_k\}$的检测中，从使差错概率最小的观点来看，DFE 不是最优的。在通过会产生 ISI 的信道传输信息的数字通信系统中，最佳检测器是一个最大似然符号序列检测器，这类检测器在它的输出端对给定接收采样序列$\{y_k\}$产生最大可能的符号序列$\{\tilde{a}_k\}$。也就是说，这个检测器找到使**似然函数**：

$$\Lambda(\{a_k\}) = \ln p(\{y_k\} \mid \{a_k\})$$

最大的序列$\{\tilde{a}_k\}$，式中 $p(\{y_k\} \mid \{a_k\})$是在$\{a_k\}$条件下的联合概率。使这个联合条件概率最大的符号序列$\{\tilde{a}_k\}$就称为**最大似然序列检测器**。

实现最大似然序列检测(MLSD)的算法是 Viterbi 算法，这个算法原本是为解卷积码(在 8.3.2 节中讨论)设计的。要了解当有 ISI 时从序列检测方面对这个算法的介绍，读者可以参

阅书末的参考文献[3,4]。

对有 ISI 的信道而言,MLSD 的主要缺点是作为 ISI 跨度的函数的计算复杂性中的指数特性。因此,MLSD 仅对 ISI 只跨几个符号并且 ISI 很严重的信道才有实际意义。因为在这种情况下会使线性均衡器或反馈判决均衡器的性能严重退化。例如,图 6.39 体现了经由信道 B(见图 6.35)传输 2 电平 PAM 信号时利用 Viterbi 算法的差错概率性能。为了进行比较,图中也画出了 DFE 的差错概率。这两个结果都是用计算机仿真得到的。可以看到,在差错概率为 10^{-4}的情况下,MLSD 的性能要比 DFE 好 4.5 dB 左右。所以这就是在相对短的 ISI 跨度的信道上 MLSD 使性能显著提高的一个例子。

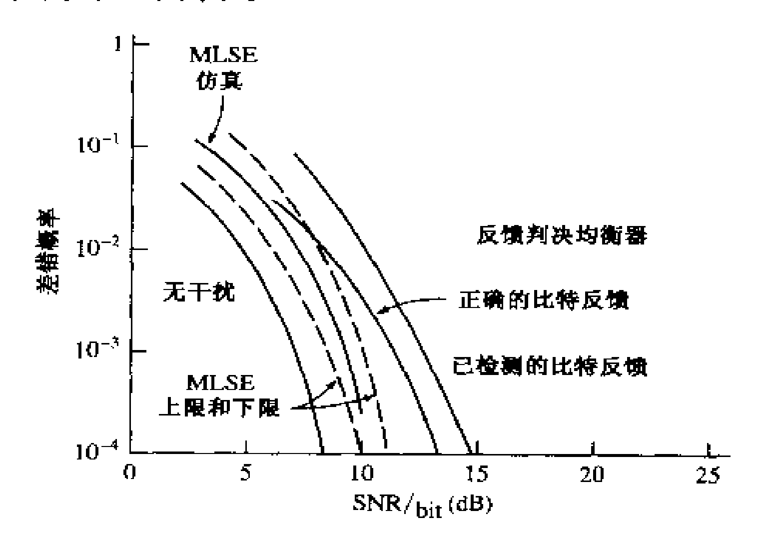


图 6.39　经由图 6.35 中的信道 B 传输 2 电平 PAM 的 Viterbi 算法的差错概率

总之,信道均衡器广泛用于数字通信系统,以降低由信道失真引起的 ISI 的影响。线性均衡器一般用于在电话信道上传输数据的高速调制解调器中。对于无线传输,诸如移动蜂窝通信和办公室之间的通信,传输信号的多径传播会形成严重的 ISI。这类信道要求功能更强的均衡器来对抗这个严重的 ISI。反馈判决均衡器和 MLSD 是两种适用于有严重 ISI 的无线信道的非线性均衡器。

6.8　习题

6.1　解说题 6.1 中的矩形脉冲的傅里叶变换和功率谱$S_v(f)$,都能用离散傅里叶变换(DFT)或 FFT 算法由 MATLAB 计算得出。归一化 $T=1$ 和 $\sigma_a^2=1$,然后在 $t=k/10, k=0,1,2,\cdots,127$ 对矩形脉冲 $g(t)$采样,得到 $g(t)$样值序列$\{g_k\}$。用 MATLAB 计算 128 点$\{g_k\}$的 DFT,并画出$|G_m|^2, m=0,1,\cdots,127$ 的值。同时也画出由式(6.2.13)给出的真正的谱$|G(f)|^2$,比较这两个结果。

6.2　$g(t)$给出为

$$g(t)=\begin{cases}\frac{1}{2}\left(1-\cos\left(\frac{2\pi t}{T}\right)\right), & 0\leqslant t\leqslant T\\ 0, & \text{其余 } t\end{cases}$$

重做习题 6.1。计算时令 $T=1$。

6.3　当脉冲 $g(t)$ 为

$$g(t)=\begin{cases}\frac{1}{2}\left(1-\cos\left(\frac{2\pi t}{T}\right)\right), & 0\leqslant t\leqslant T\\ 0, & \text{其余 } t\end{cases}$$

并且信号幅度序列 $\{a_n\}$ 有式(6.2.14)给出的相关函数时，写一个 MATLAB 程序计算信号 $v(t)$ 的功率谱 $S_v(f)$。

6.4　利用 MATLAB 设计一个 FIR 线性相位滤波器对某个低通带限信道建模，该信道在 $|f|\leqslant$ 3000 Hz 的通带内有 $\frac{1}{2}$ dB 波纹(起伏)，在 $|f|\geqslant$3500 Hz 的阻带有 −40 dB 的衰减。画出冲激响应和频率响应。

6.5　写一个 MATLAB 程序，设计一个 FIR 线性相位滤波器，对某个低通带限信道建模，该信道有如下期望的幅度响应：

$$A(f)=\begin{cases}1, & |f|\leqslant 3000\\ 0, & f>3000\end{cases}$$

设计采用窗函数法，利用 Hanning 窗。

6.6　写一个 MATLAB 程序，该程序产生在解说题 6.5 中的两径(多径)信道的冲激响应，并画出 $p=0.95$ 和 5 个样本延时的冲激响应。

6.7　写一个 MATLAB 仿真程序，实现解说题 6.6 中的信道 1，并当 10 000 个二进制数据比特 $\{\pm 1\}$ 经由该信道传输时测量误码率。信道受到方差为 $\sigma^2=0$，$\sigma^2=0.1$，$\sigma^2=0.2$，$\sigma^5=0.5$ 和 $\sigma^2=1.0$ 的 AWGN 污损。

6.8　若信道为

$$x_n=\begin{cases}1, & n=0\\ 0.25, & n=\pm 1\\ 0, & \text{其余 } n\end{cases}$$

重做习题 6.7。

6.9　写一个 MATLAB 程序，产生具有任意滚降系数 α，由式(6.5.9)给出的发送滤波器冲激响应 $g_T(t)$ 的样本，求出并画出 $\alpha=\frac{1}{2}$ 和 $N=31$ 时的 $g_T(n)$。

同时也求出并画出这个滤波器的频率响应特性的幅度特性(通过将 $g_T(n)$ 补上 $3N$ 个零值，再求 $g_T(n)$ 的 $4N$ 点的 DFT]。

6.10　写一个 MATLAB 程序，计算任何发送滤波器 $g_T(n)$ 与接收端和其匹配的滤波器级联的总冲激响应。这个计算可用 DFT 按下述步骤进行：将 $g_T(n)$ 补 $N-1$(或更多)个零值并计算$(2N-1)$点(或更多)的 DFT，得到 $G_T(k)$。然后形成 $|G_T(k)|^2$ 并计算 $|G_T(k)|^2$ 的$(2N-1)$点的逆 DFT。对习题 6.9 中的这个滤波器求总冲激响应，并将这一结果与以 $F_s=4/T$ 采样 $x_{rc}(t)$ 所得的理想冲激响应进行比较。

6.11　对于 $N=21$ 和 $N=41$，重做习题 6.9。画出这个离散时间滤波器的频率响应，并将它与在习题 6.9 中所得的结果进行比较。请读者指出主要差别。

6.12　写一个 MATLAB 程序，它取某个数据序列 $\{D_k\}$，将它对一个修正双二进制脉冲传输系统进行预编码以产生 $\{p_k\}$，并将这个预编码序列映射为发送幅度电平 $\{a_k\}$。然后，由这

个发送序列形成无噪声的接收序列$\{b_k = a_k - a_{k-2}\}$，并利用式(6.5.37)的关系恢复数据信号$\{D_k\}$。对$M=2$和$M=4$的发送幅度电平的任意伪随机数据序列$\{D_k\}$运行这个程序，并检查结果。

6.13 写一个 MATLAB 程序，对采用双二进制信号脉冲的 2 电平 PAM 通信系统进行 Monte Carlo 仿真，这里的预编码和幅度序列$\{a_k\}$都按解说题 6.9 完成。按式(6.5.22)指出的，将高斯噪声加到接收序列$\{b_k\}$以形成检测器的输入，并用式(6.5.28)的检测规则恢复这个数据。对 10 000 个比特执行这个仿真，并在$\sigma^2=0.1$，$\sigma^2=0.5$和$\sigma^2=1$时测量比特差错概率。画出无 ISI 的 2 电平 PAM 的理论差错概率，并将 Monte Carlo 的仿真结果与该理想性能进行比较。应该能够看到，在双二进制系统性能上有某些小的降低。

6.14 对采样率$F_s=8/T$，$\alpha=\frac{1}{2}$和$N=61$，重做习题 6.9。高的采样率会得到更好的频率响应特性吗？也就是说与$X_{rc}(f)$更匹配吗？

6.15 对于习题 6.14 设计的滤波器，利用在习题 6.10 中叙述的过程计算并画出这个滤波器和其匹配滤波器级联的输出。将这个采样的冲激响应与用采样率$F_s=8/T$对$x_{rc}(t)$采样得到的理想冲激响应进行比较。这个较高的采样率产生了这个离散时间滤波器的冲激响应对理想滤波器冲激响应的一个更好的近似吗？

6.16 写一个 MATLAB 程序，该程序针对由式(6.5.18)给出的修正双二进制脉冲产生由式(6.5.21)给出的发送滤波器的冲激响应$g_T(t)$的采样。求出并画出$N=31$时的$g_T(n)$。同时还要求出并画出这个滤波器的频率响应的幅度。

6.17 对习题 6.16 设计的滤波器重做习题 6.10。

6.18 对$N=21$和$N=41$，重做习题 6.16。将这两个滤波器的频率响应与习题 6.16 设计的滤波器的频率响应进行比较。这些频率响应特性的主要差别是什么？

6.19 考虑解说题 6.10 中的信道失真脉冲$x(t)$，该脉冲以$2/T$采样率采样并用$2K+1=11$节的强迫归零均衡器进行均衡。写一个 MATLAB 程序来解这些强迫归零均衡器的系数，求出并画出这个归零均衡器 50 个采样值的输出。

6.20 对于 MSE 均衡器，用$N_0=0.01$，$N_0=0.1$和$N_0=1.0$重做习题 6.19。将这些均衡器的系数与习题 6.19 求得的系数进行比较，并当N_0改变时讨论这个结果。

6.21 写一个 MATLAB 程序，计算基于 MSE 准则的任意长度$2K+1$的 FIR 均衡器的抽头系数，已知在符号率下取得的脉冲$x(t)$的采样值和加性噪声的谱密度N_0作为输入，用这个程序求当$x(t)$的采样值为

$$x(nT)=\begin{cases}1, & n=0\\ 0.5, & n=\pm 1\\ 0.3, & n=\pm 3\\ 0.1, & n=\pm 4\end{cases}$$

并且$N_0=0.01$，$N_0=0.1$时 11 节均衡器的系数。同时还要求出最佳均衡器系数的最小 MSE。

6.22 对习题 6.21 给出的信道特性，当均衡器抽头数为 21 时求 MSE 均衡器系数和最小 MSE。将这些系数与习题 6.21 所得的系数值进行比较，并讨论用较长的均衡器获得的在 MSE

上的减小是否足够大,以证实它的应用是恰当的。

6.23　将信道采样的响应与均衡器系数求卷积,并观察所产生的输出序列,可以求出在均衡器输出中残余的 ISI 量。写一个 MALTAB 程序,当输入是采样的信道特性时,计算一个给定长度的均衡器的输出。为简单起见,考虑均衡器是符号间隔均衡器的情况,信道采样响应也由按符号间隔样本组成。用这个程序求习题 6.21 给出的信道响应的一个归零符号间隔均衡器的输出。

6.24　写一个 MATLAB Monte Carlo 仿真程序,仿真图 P6.24 中建模的数字通信系统。信道按符号间隔值的 FIR 滤波器建模,MSE 均衡器也是一个符号间隔抽头系数 FIR 滤波器。用发送训练符号对该均衡器进行初始训练。在数据方式中,该均衡器采用检测器的输出形成误差信号。利用 1000 个训练(二进制)符号和 10 000 个二进制数据符号对习题 6.21给出的信道模型进行这个系统的 Monte Carlo 仿真。分别采用的是 $N_0=0.01$, $N_0=0.1$ 和 $N_0=1$。将测出的误码率与无 ISI 的理想信道误码率进行比较。

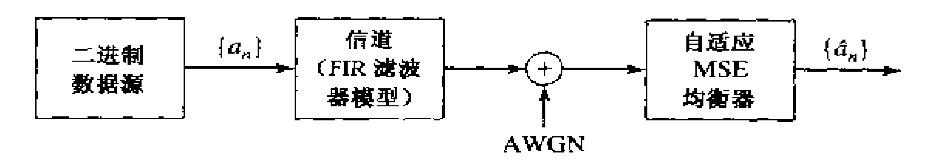


图 P6.24

第 7 章　载波调制的数字传输

7.1　概述

前面两章讨论的都是数字信息经由基带的传输。在这种情况下，载有信息的信号是直接通过信道传输的，而不用某个正弦载波。然而，大多数通信信道都是带通信道，因此通过这类信道传输信号的惟一办法是将载有信息的信号频率搬移到信道的频带之内。

第 7 章将要讨论适合于带通信道的载波调制信号的 4 种形式，包括：幅度调制信号、正交幅度调制信号、相移键控和频移键控。

7.2　载波幅度调制

在基带数字 PAM 中，信号波形具有如下形式：

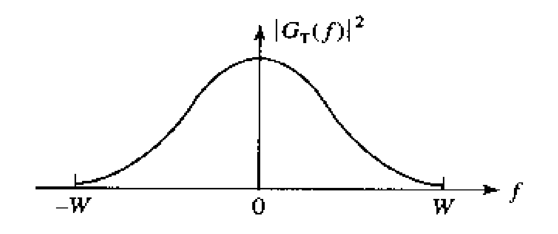
$$s_m(t) = A_m g_T(t) \tag{7.2.1}$$

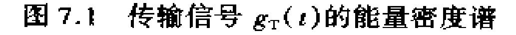
其中，A_m 是第 m 个波形的幅度，$g_T(t)$是某一种脉冲，它的形状决定了传输信号的谱特性。假设基带信号的频谱位于频带$|f| \leqslant W$之内，W 是$|G_T(f)|^2$ 的带宽，如图 7.1 所示。我们已经知道，这个信号幅度取的是离散值：

$$A_m = (2m-1-M)d, \quad m = 1,2,\cdots,M \tag{7.2.2}$$

其中，$2d$ 是两相邻信号点之间的欧氏距离。

为了通过一个带通信道传输这个数字信号波形，就要将这个基带信号波形 $s_m(t)$，$m=1,2,\cdots,M$ 乘以形如 $\cos(2\pi f_c t)$的正弦载波，如图 7.2 所示，其中 f_c 是载波频率($f_c > W$)，对应于信道通带的中心频率。因此，传输的信号波形表示为：




图 7.1　传输信号 $g_T(t)$的能量密度谱

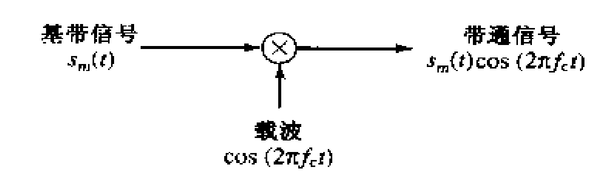


图 7.2　采用基带 PAM 信号的正弦载波幅度调制

$$u_m(t) = A_m g_T(t)\cos(2\pi f_c t), \quad m = 1,2,\cdots,M \tag{7.2.3}$$

在传输的脉冲形状 $g_T(t)$是矩形的特殊情况下，即

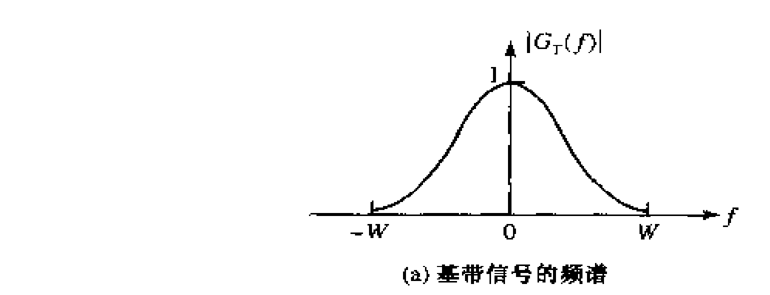
$$g_T(t) = \begin{cases} \sqrt{\dfrac{2}{T}}, & 0 \leqslant t \leqslant T \\ 0, & \text{其余 } t \end{cases}$$

这个幅度已调的载波信号通常称为**幅度键控**(amplitude-shift keying,简称 ASK)。在这种情况下,这个 PAM 信号不是带限的。

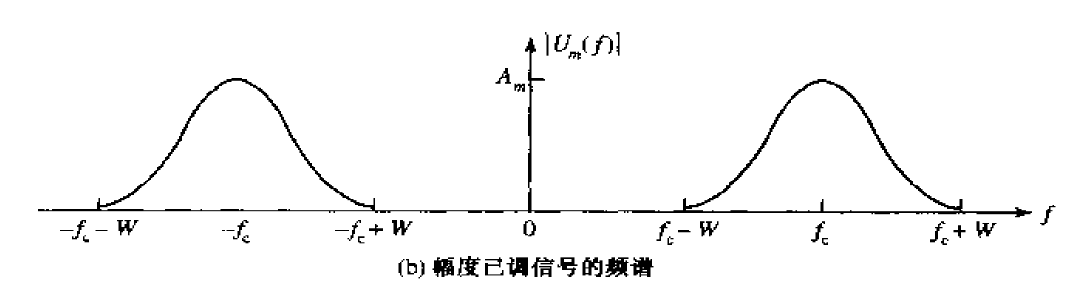
用基带信号波形 $s_m(t)$的 $\cos(2\pi f_c t)$载波幅度调制,将基带信号的频谱搬移一个 f_c 的量,并由此将信号放在信道的带宽内。回忆一下,载波的傅里叶变换是$[\delta(f-f_c)+\delta(f+f_c)]/2$,又由于两个信号在时域内相乘就相应于在频域的频谱的卷积,所以这个幅度已调信号的频谱是:

$$U_m(f)=\frac{A_m}{2}[G_T(f-f_c)+G_T(f+f_c)] \tag{7.2.4}$$

因此,基带信号 $s_m(t)=A_m g_T(t)$的频谱就被搬移到载波频率 f_c 的频率上。这个带通信号是一个双边带载波抑制(DSB-SC)的 AM 信号,如图 7.3 所示。



(a) 基带信号的频谱



(b) 幅度已调信号的频谱

图 7.3 幅度调制

值得注意的是,将基带信号 $s_m(t)$"放"在载波信号 $\cos(2\pi f_c t)$的幅度上,并没有改变这个数字 PAM 信号波形的基本几何表示。一般来说,这个带通 PAM 信号波形可以表示成:

$$u_m(t)=s_m\psi(t) \tag{7.2.5}$$

这里定义信号波形 $\psi(t)$为

$$\psi(t)=g_T(t)\cos(2\pi f_c t) \tag{7.2.6}$$

和

$$s_m=A_m,\qquad m=1,2,\cdots,M \tag{7.2.7}$$

代表在实线上取 M 个值的信号点,如图 7.4 所示。

$-5d \quad -3d \quad -d \quad 0 \quad d \quad 3d \quad 5d$

图 7.4 PAM 信号的信号点星座图

将信号波形 $\psi(t)$归一化到单位能量,即

$$\int_{-\infty}^{\infty}\psi^2(t)\mathrm{d}t=1 \tag{7.2.8}$$

这样就有：

$$\int_{-\infty}^{\infty} g_T^2(t)\cos^2(2\pi f_c t)\mathrm{d}t = \frac{1}{2}\int_{-\infty}^{\infty} g_T^2(t)\mathrm{d}t + \frac{1}{2}\int_{-\infty}^{\infty} g_T^2(t)\cos(4\pi f_c t)\mathrm{d}t = 1 \quad (7.2.9)$$

但是,由于 $g_T(t)$的带宽 W 比载波频率小得多,即 $f_c >> W$,所以有：

$$\int_{-\infty}^{\infty} g_T^2(t)\cos(4\pi f_c t)\mathrm{d}t = 0 \quad (7.2.10)$$

在这种情况下,在 $\cos(4\pi f_c t)$的任何一个周期内,$g_T(t)$基本上是不变的,所以式(7.2.10)在被积函数的每个周期,积分都是零。由于式(7.2.10),可得：

$$\frac{1}{2}\int_{-\infty}^{\infty} g_T^2(t)\mathrm{d}t = 1 \quad (7.2.11)$$

因此,$g_T(t)$必须适当地加权,以使式(7.2.8)和式(7.2.11)都满足。

7.2.1 PAM 信号的解调

一个带通数字 PAM 信号的解调可以用相关或匹配滤波的几种方式之一来完成。为了解释方便,现考虑一种相关型式的解调器。

接收信号可表示为：

$$r(t) = A_m g_T(t)\cos(2\pi f_c t) + n(t) \quad (7.2.12)$$

其中,$n(t)$是某个带通噪声过程,它可以表示为：

$$n(t) = n_c(t)\cos(2\pi f_c t) - n_s(t)\sin(2\pi f_c t) \quad (7.2.13)$$

其中,$n_c(t)$和 $n_s(t)$是该噪声的正交分量。通过将接收信号 $r(t)$和由式(7.2.6)给出的 $\psi(t)$做互相关,如图 7.5 所示,可得输出为：

$$\int_{-\infty}^{\infty} r(t)\psi(t)\mathrm{d}t = A_m + n = s_m + n \quad (7.2.14)$$

其中 n 代表在相关器输出中的加性噪声分量。

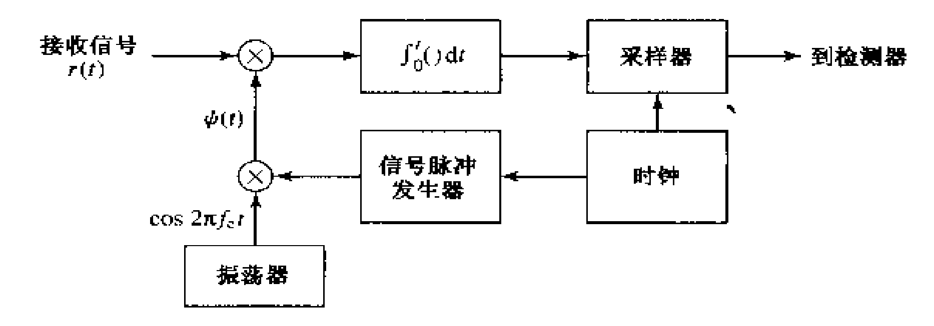


图 7.5　带通数字 PAM 信号的解调

这个噪声分量具有零均值,它的方差可以表示为：

$$\sigma_n^2 = \int_{-\infty}^{\infty} |\psi(f)|^2 S_n(f)\mathrm{d}f \quad (7.2.15)$$

其中,$\psi(f)$是 $\psi(t)$的傅里叶变换,$S_n(f)$是加性噪声的功率谱密度。$\psi(t)$的傅里叶变换是：

$$\psi(f) = \frac{1}{2}[G_T(f-f_c) + G_T(f+f_c)] \quad (7.2.16)$$

这个带通加性噪声过程的功率谱密度是：

$$\mathcal{S}_n(f)=\begin{cases}\dfrac{N_0}{2}, & |f-f_c|\leqslant W\\ 0, & \text{其余 } f\end{cases} \tag{7.2.17}$$

将式(7.2.16)和式(7.2.17)代入式(7.2.15),并求出该积分值,可得 $\sigma_n^2=N_0/2$。

由式(7.2.14)(它就是幅度检测器的输入)显然可见,载波调制 PAM 信号的最佳检测器的差错概率与基带 PAM 的最佳检测差错概率是一样的,即:

$$P_M=\frac{2(M-1)}{M}Q\left(\sqrt{\frac{6(\log_2 M)\mathcal{E}_{\text{avb}}}{(M^2-1)N_0}}\right) \tag{7.2.18}$$

其中,$\mathcal{E}_{\text{avb}}$是每比特的平均能量。

解说题

解说题 7.1　[PAM 信号频谱]

在某一幅度调制数字 PAM 系统中,发送滤波器具有开根升余弦频谱特性的冲激响应 $g_T(t)$,如解说题 6.7 所述,滚降系数 $\alpha=0.5$。载波频率 $f_c=40/T$。求出并画出该基带信号的频谱和幅度已调信号的频谱。

题　解

图 7.6 给出了这两个信号的频谱特性。

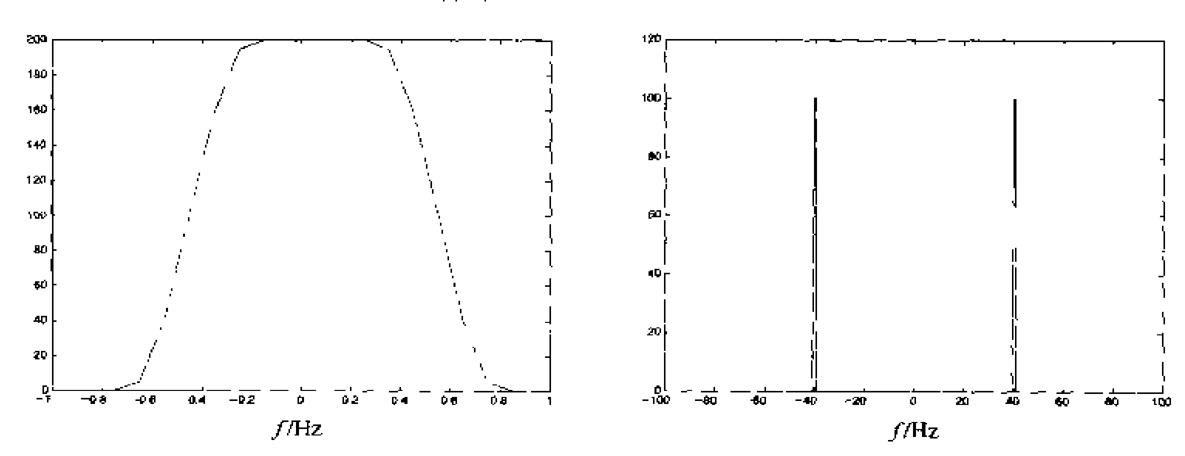


图 7.6　基带信号和幅度已调(带通)信号的频谱

计算本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrated Problem 7.1.
echo on
T=1;
delta_T=T/200;                    % sampling interval
alpha=0.5;                        % rolloff factor
fc=40/T;                          % carrier frequency
A_m=1;                            % amplitude
t=-5*T+delta_T:delta_T:5*T;       % time axis
N=length(t);
for i=1:N,
```

```
  if (abs(t(i))~=T/(2*alpha)),
    g_T(i) = sinc(t(i)/T)*(cos(pi*alpha*t(i)/T)/(1-4*alpha^2*t(i)^2/T^2));
  else
    g_T(i) = 0;                        % The value of g_T is 0 at t=T/(2*alpha)
  end;                                 % and at t=-T/(2*alpha).
  echo off ;
end;
echo on;
G_T=abs(fft(g_T));                     % spectrum of g_T
u_m=A_m*g_T.*cos(2*pi*fc*t);           % the modulated signal
U_m=abs(fft(u_m));                     % spectrum of the modulated signal
% actual frequency scale
f=-0.5/delta_T:1/(delta_T*(N-1)):0.5/delta_T;
% Plotting commands follow.
figure(1);
plot(f,fftshift(G_T));
axis([-1/T 1/T 0 max(G_T)]);
figure(2);
plot(f,fftshift(U_m));
```

7.3 载波相位调制

在载波相位调制中，要经过一条通信信道传输的信息寄寓在载波的相位中。由于载波相位的范围是 $0 \leqslant \theta \leqslant 2\pi$，因此经由数字相位调制用于传输数字信息的载波相位就是：$\theta_m = 2\pi m/M$，$m = 0, 1, \cdots, M-1$。这样，对于二相相位调制（$M=2$）来说，两个载波相位是 $\theta_0 = 0$ 和 $\theta_1 = \pi$。对于 M 相相位调制来说，$M = 2^k$，其中 k 是每个传输符号的信息比特数。

一组 M 载波相位调制信号波形的一般表示式为：

$$u_m(t) = Ag_T(t)\cos\left(2\pi f_c t + \frac{2\pi m}{M}\right), \qquad m = 0, 1, \cdots, M-1 \tag{7.3.1}$$

其中，$g_T(t)$是发送滤波器的脉冲形状，它决定了传输信号的频谱特性，A 是信号幅度。这类数字相位调制称为相移键控（phase-shift keying，简称 PSK）。注意，PSK 信号对所有 m 都具有相等的能量，即

$$\mathcal{E}_m = \int_{-\infty}^{\infty} u_m^2(t)\mathrm{d}t \tag{7.3.2}$$

$$= \int_{-\infty}^{\infty} A^2 g_T^2(t)\cos^2\left(2\pi f_c t + \frac{2\pi m}{M}\right)\mathrm{d}t$$

$$= \frac{1}{2}\int_{-\infty}^{\infty} A^2 g_T^2(t)\mathrm{d}t + \frac{1}{2}\int_{-\infty}^{\infty} A^2 g_T^2(t)\cos\left(4\pi f_c t + \frac{4\pi m}{M}\right)\mathrm{d}t$$

$$= \frac{A^2}{2}\int_{-\infty}^{\infty} g_T^2(t)\mathrm{d}t \tag{7.3.3}$$

$$= \mathcal{E}_s, \quad \text{全部 } m \tag{7.3.4}$$

其中，$\mathcal{E}_s$ 代表每个传输符号的能量。在式(7.3.2)中，凡涉及双倍载波频率的项在 $f_c >> W$ 时积分后都为零，W 是 $g_T(t)$的带宽。

当 $g_T(t)$是一个矩形脉冲时，定义为：

$$g_T(t)=\sqrt{\frac{2}{T}},\quad 0\leqslant t\leqslant T \tag{7.3.5}$$

这时在符号区间 $0\leqslant t\leqslant T$ 内传输的信号波形可以表示为(用 $A=\sqrt{\mathscr{E}_s}$):

$$u_m(t)=\sqrt{\frac{2\mathscr{E}_s}{T}}\cos\left(2\pi f_c t+\frac{2\pi m}{M}\right),\quad m=0,1,\cdots,M-1 \tag{7.3.6}$$

注意,由式(7.3.6)给出的传输信号有一个不变的包络,而载波相位则在每个信号区间开始时突然变化。图 7.7 说明了一种 4 相($M=4$)PSK 的信号波形。

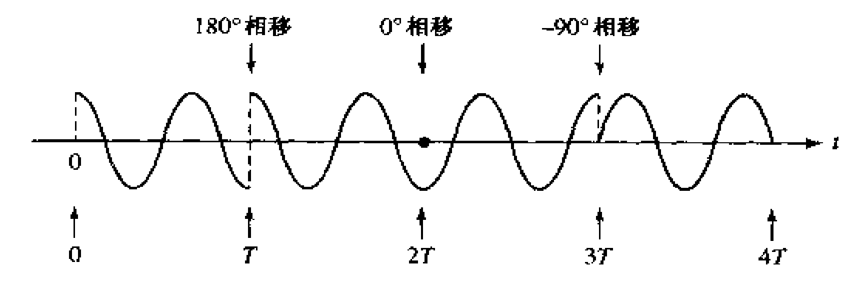


图 7.7 4 相 PSK 信号举例

把式(7.3.6)中余弦函数的相角看成两个相角的和,即可将式(7.3.1)的波形表示为:

$$\begin{aligned}u_m(t)&=\sqrt{\mathscr{E}_s}\,g_T(t)\cos\left(\frac{2\pi m}{M}\right)\cos(2\pi f_c t)-\sqrt{\mathscr{E}_s}\,g_T(t)\sin\left(\frac{2\pi m}{M}\right)\sin(2\pi f_c t)\\&=s_{mc}\psi_1(t)+s_{ms}\psi_2(t)\end{aligned} \tag{7.3.7}$$

其中

$$\begin{aligned}s_{mc}&=\sqrt{\mathscr{E}_s}\cos\frac{2\pi m}{M}\\s_{ms}&=\sqrt{\mathscr{E}_s}\sin\frac{2\pi m}{M}\end{aligned} \tag{7.3.8}$$

而 $\psi_1(t)$ 和 $\psi_2(t)$ 是两个正交基函数,定义为:

$$\begin{aligned}\psi_1(t)&=g_T(t)\cos(2\pi f_c t)\\\psi_2(t)&=-g_T(t)\sin(2\pi f_c t)\end{aligned} \tag{7.3.9}$$

通过适当地将脉冲波形 $g_T(t)$ 归一化,就可以将这两个基函数的能量归一化到 1。这样,一个相位调制信号可以看成两个正交载波,其幅度取决于在每个信号区间内传输的相位。因此,数字相位调制信号从几何上可用分量为 s_{mc} 和 s_{ms} 的二维向量来表示,即

$$s_m=\left(\sqrt{\mathscr{E}_s}\cos\frac{2\pi m}{M}\ \sqrt{\mathscr{E}_s}\sin\frac{2\pi m}{M}\right) \tag{7.3.10}$$

图 7.8 给出了 $M=2,4$ 和 8 的信号点星座图。我们可以看到,二相相位调制与 2 电平 PAM(二进制反极性信号)是相同的。

将 k 个信息比特映射(或分配)到 $M=2^k$ 个可能的相位上,可以用几种方法来完成。其中,优先考虑使用**格雷编码**(Gray encoding)。在这种编码中,相邻相位只差 1 个二进制比特,如图 7.8 所示。结果当噪声对传输相位引起相邻相位的错误选取时,在用格雷编码的 k 比特序列中仅会产生一个单一的比特差错。

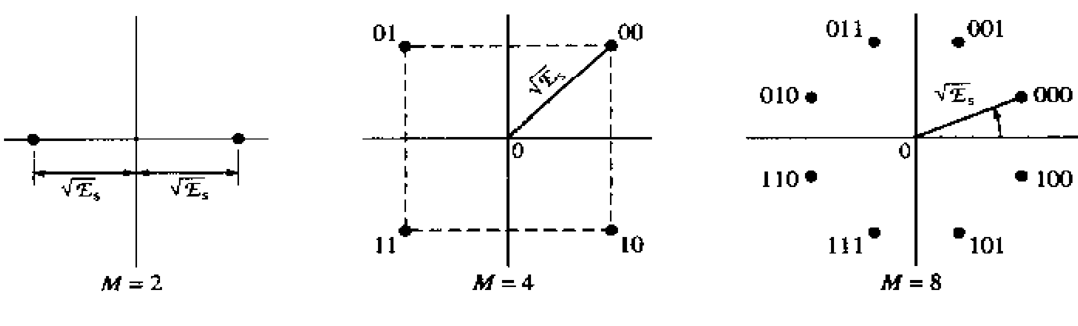


图 7.8 PSK 信号星座图

解说题

解说题 7.2 [PSK 波形]

对 $M=8$,产生由式(7.3.6)给出的恒定包络的 PSK 信号波形。为简单起见,信号幅度归一化到 1。

题 解

图 7.9 给出了 $f_c=6/T$ 时的 8 个波形。

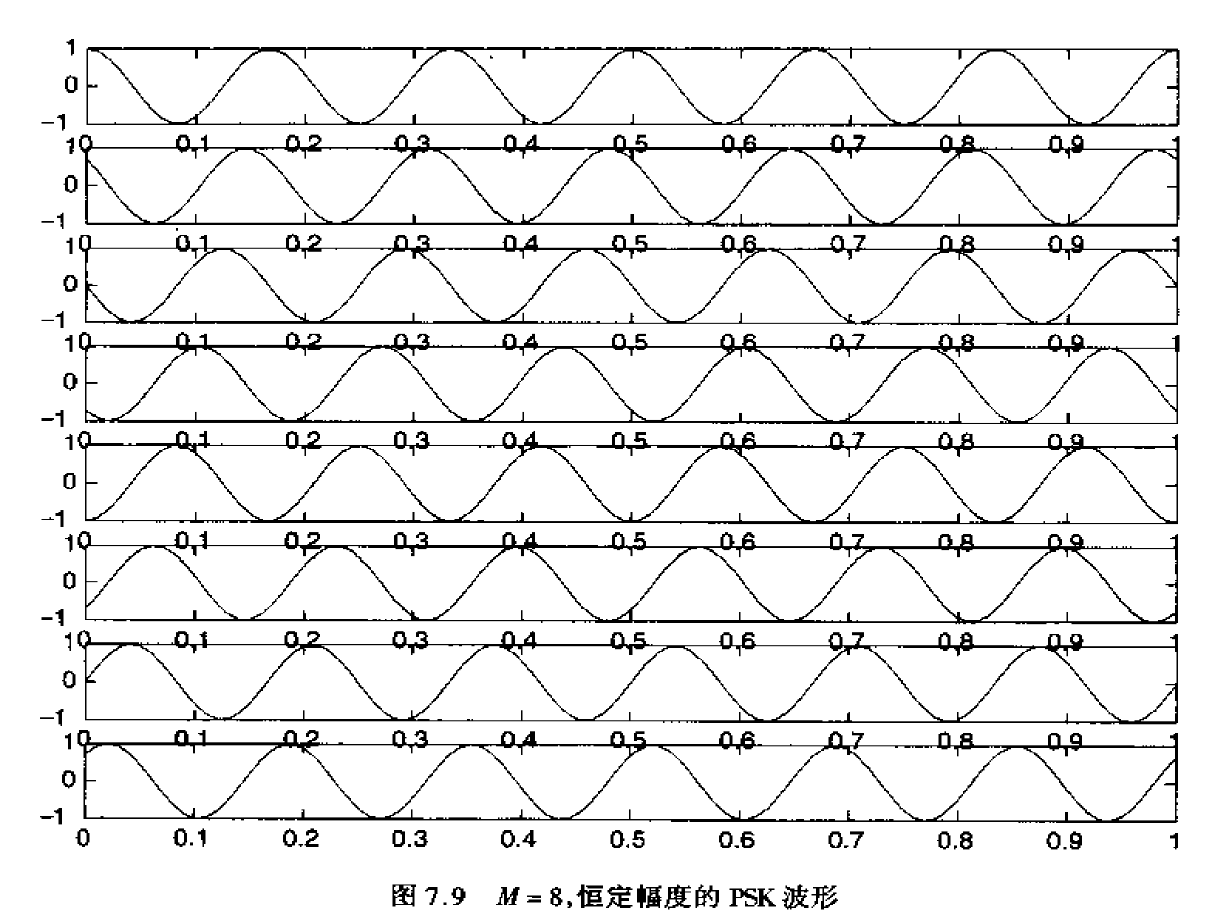


图 7.9 $M=8$,恒定幅度的 PSK 波形

计算本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 7.2.
echo on
T=1;
M=8;
Es=T/2;
fc=6/T;                              % carrier frequency
N=100;                               % number of samples
delta_T=T/(N-1);
t=0:delta_T:T;
u0=sqrt(2*Es/T)*cos(2*pi*fc*t);
u1=sqrt(2*Es/T)*cos(2*pi*fc*t+2*pi/M);
u2=sqrt(2*Es/T)*cos(2*pi*fc*t+4*pi/M);
u3=sqrt(2*Es/T)*cos(2*pi*fc*t+6*pi/M);
u4=sqrt(2*Es/T)*cos(2*pi*fc*t+8*pi/M);
u5=sqrt(2*Es/T)*cos(2*pi*fc*t+10*pi/M);
u6=sqrt(2*Es/T)*cos(2*pi*fc*t+12*pi/M);
u7=sqrt(2*Es/T)*cos(2*pi*fc*t+14*pi/M);
% plotting commands follow
subplot(8,1,1);
plot(t,u0);
subplot(8,1,2);
plot(t,u1);
subplot(8,1,3);
plot(t,u2);
subplot(8,1,4);
plot(t,u3);
subplot(8,1,5);
plot(t,u4);
subplot(8,1,6);
plot(t,u5);
subplot(8,1,7);
plot(t,u6);
subplot(8,1,8);
plot(t,u7);
```

7.3.1 相位解调与检测

从 AWGN 信道中,在一个信号区间内接收到的带通信号可以表示为:

$$\begin{aligned} r(t) &= u_m(t) + n(t) \\ &= u_m(t) + n_c(t)\cos(2\pi f_c t) - n_s(t)\sin(2\pi f_c t) \end{aligned} \tag{7.3.11}$$

其中,$n_c(t)$和 $n_s(t)$是加性噪声的两个正交分量。

可以将这个接收信号与由式(7.3.9)给出的 $\psi_1(t)$和 $\psi_2(t)$做相关,两个相关器的输出产生受噪声污损的信号分量,它们可以表示为:

$$\begin{aligned} \boldsymbol{r} &= \boldsymbol{s}_m + \boldsymbol{n} \\ &= \left(\sqrt{\mathcal{E}_s}\cos\frac{2\pi m}{M} + n_c \quad \sqrt{\mathcal{E}_s}\sin\frac{2\pi m}{M} + n_s\right) \end{aligned} \tag{7.3.12}$$

其中,n_c 和 n_s 定义为:

$$n_c = \frac{1}{2}\int_{-\infty}^{\infty} g_T(t) n_c(t)\mathrm{d}t$$

$$n_s = \frac{1}{2}\int_{-\infty}^{\infty} g_T(t) n_s(t) \mathrm{d}t \tag{7.3.13}$$

这两个正交噪声分量 $n_c(t)$和 $n_s(t)$是零均值且互不相关的高斯随机过程。这样,$E(n_c) = E(n_s) = 0$ 并且 $E(n_c n_s) = 0$。n_c 和 n_s 的方差是:

$$E(n_c^2) = E(n_s^2) = \frac{N_0}{2} \tag{7.3.14}$$

最佳检测器将接收信号向量 $\boldsymbol{r}$ 投射到 M 个可能的传输信号向量$\{\boldsymbol{s}_m\}$之一上,并选取对应于最大投影的向量。据此,得到相关准则为:

$$C(\boldsymbol{r}, \boldsymbol{s}_m) = \boldsymbol{r} \cdot \boldsymbol{s}_m, \quad m = 0, 1, \cdots, M-1 \tag{7.3.15}$$

由于全部信号都具有相等的能量,因此数字相位调制的一种等效检测器标准是计算接收信号向量 $\boldsymbol{r} = (\boldsymbol{r}_c, \boldsymbol{r}_s)$的相位为:

$$\theta_r = \arctan \frac{r_s}{r_c} \tag{7.3.16}$$

并从信号集$\{s_m\}$中选取其相位最接近 θ_r 的信号。

在 AWGN 信道中,相位调制在检测器端的差错概率可以在有关数字通信的任何教材中找到。因为二相(或二元)相位调制与二进制 PAM 是相同的,所以差错概率为:

$$P_2 = Q\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right) \tag{7.3.17}$$

其中,$\mathcal{E}_b$ 是每比特的能量。四相相位调制可以看成是两个在正交载波上的二相相位调制系统,所以 1 个比特的差错概率与二相相位调制是一样的。对于 $M > 4$ 的符号差错概率,不存在简单的闭式表达式。对 P_M 的一种好的近似式是:

$$\begin{aligned} P_M &\approx 2Q\left(\sqrt{\frac{2\mathcal{E}_s}{N_0}} \sin\frac{\pi}{M}\right) \\ &\approx 2Q\left(\sqrt{\frac{2k\mathcal{E}_b}{N_0}} \sin\frac{\pi}{M}\right) \end{aligned} \tag{7.3.18}$$

其中,$k = \log_2 M$ 比特/符号。图 7.10 给出了作为 SNR($\mathcal{E}_b/N_0$)的函数的符号概率特性。

由于将 k 比特符号映射到对应的信号相位的多种可能性,所以对 M 相相位调制来说,等效的比特差错概率也很难导出。但是,当在映射中使用格雷编码时,对应于相邻信号相位的两个 k 比特符号仅在某个单一比特有不同;而由噪声引起的最大可能差错是将一个相邻相位错误地选为真正的相位,所以最大的比特符号差错仅包含单个比特差错。因此,对 M 相相位调制的等效比特差错概率能很好地由下式给予近似:

$$P_b \approx \frac{1}{k} P_M \tag{7.3.19}$$

解说题 7.3 [PSK 仿真]

完成 $M = 4$ 的 PSK 通信系统的 Monte Carlo 仿真,该系统计算式(7.3.15)给出的相关准则,进行检测器建模,待仿真的系统的模型如图 7.11 所示。

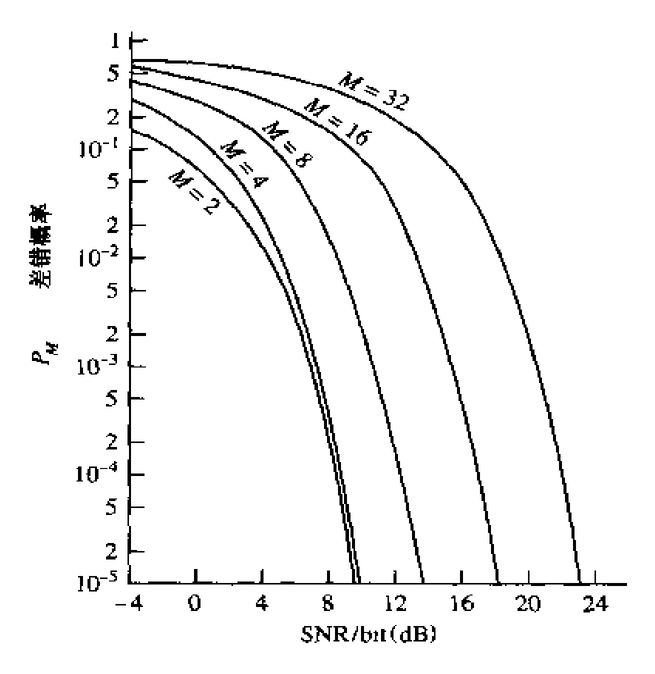


图 7.10 M 相 PSK 的一个符号差错概率

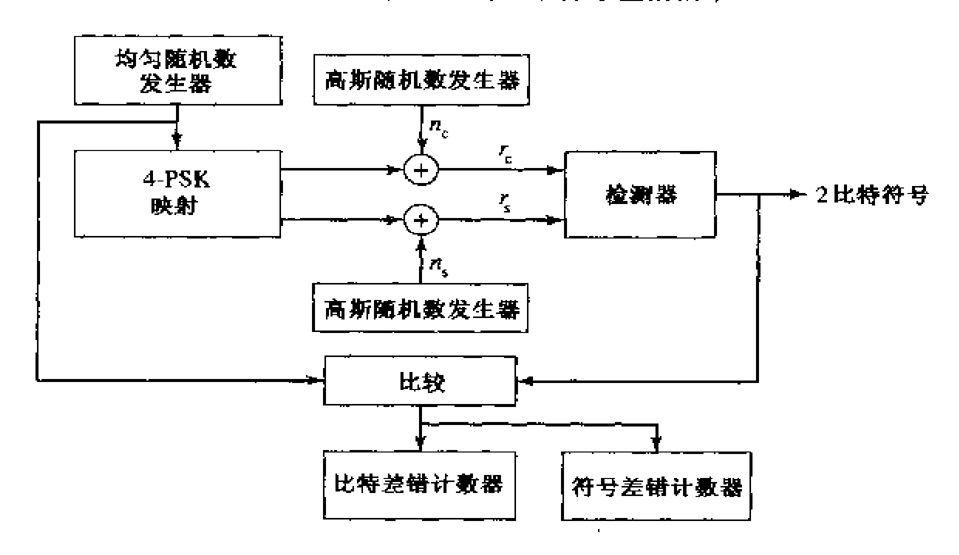


图 7.11 用于 Monte Carlo 仿真的 $M=4$ 的 PSK 系统的方框图

题 解

如图 7.11 所示，我们要仿真由式(7.3.12)给出的随机向量 $\boldsymbol{r}$ 的产生，它是信号相关器的输出和检测器的输入。先产生一个 4 种符号(2 比特)的序列，将它映射到相应的 4 相信号点，如图 7.8 中 $M=4$ 的情况所示。为了完成这个任务，利用一个随机数发生器，它会产生(0,1)范围内的均匀随机数。再将这个范围分成 4 个相等的区间(0,0.25)，(0.25,0.5)，(0.5,0.75)和(0.75,1.0)，这些子区间分别对应于 00,01,11 和 10 信息比特对，再用这些比特对来选择信号相位向量 s_m。

加性噪声分量 n_c 和 n_s 都是统计独立的零均值、方差为 σ^2 的高斯随机变量。为简单起见，可以将方差归一化到 $\sigma^2=1$，而通过给信号能量参数$\mathcal{E}_s$加权来控制接收信号中的 SNR。反之亦然。

检测器观察到接收信号向量 $\mathbf{r} = \mathbf{s}_m + \mathbf{n}$，由式(7.3.12)给出，并计算 $\mathbf{r}$ 在 4 种可能的信号向量 $\mathbf{s}_m$ 上的投影(点乘)。根据选取对应于最大投影的信号点做判决，将检测器的输出判决与传输符号进行比较，最后对符号差错和比特差错计数。

图 7.12 给出了对于不同的 SNR 参数$\mathcal{E}_b/N_0$，其中$\mathcal{E}_b = \mathcal{E}_s/2$ 是比特能量，传输 10 000 个符号的 Monte Carlo 仿真结果。图 7.12 中所示的是比特误码率，定义为 $P_b \approx P_M/2$，对应的理论差错概率由式(7.3.18)给出。

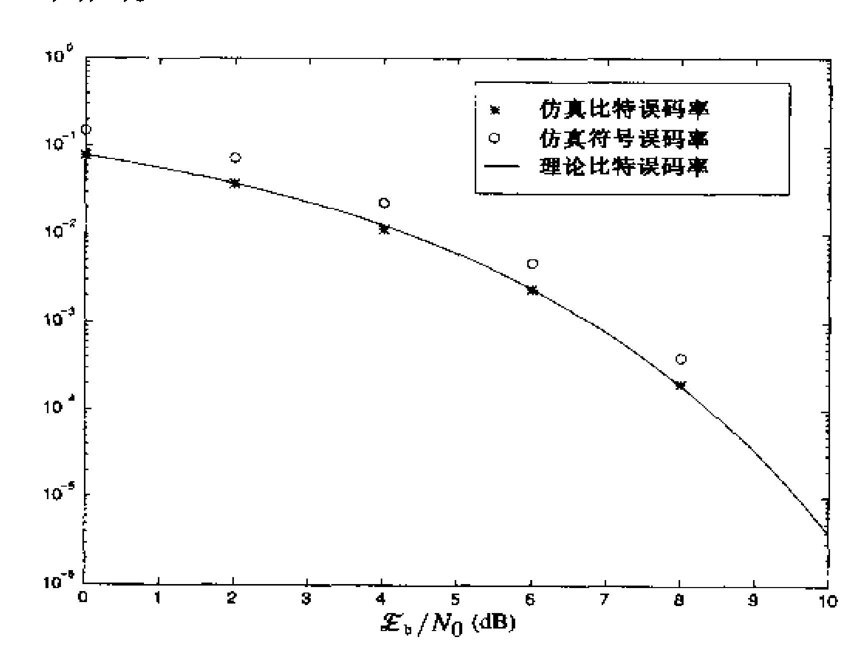


图 7.12　由 Monte Garlo 仿真得出的四相 PSK 系统的性能

计算这个 Monte Carlo 仿真的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 7.3.
echo on
SNRindB1=0:2:10;
SNRindB2=0:0.1:10;
for i=1:length(SNRindB1),
  [pb,ps]=cm_sm32(SNRindB1(i));                % simulated bit and symbol error rates
  smld_bit_err_prb(i)=pb;
  smld_symbol_err_prb(i)=ps;
  echo off ;
end;
echo on;
for i=1:length(SNRindB2),
  SNR=exp(SNRindB2(i)*log(10)/10);             % signal-to-noise ratio
  theo_err_prb(i)=Qfunct(sqrt(2*SNR));         % theoretical bit-error rate
  echo off ;
end;
echo on ;
```

```
% Plotting commands follow.
semilogy(SNRindB1,smld_bit_err_prb,'*');
hold
semilogy(SNRindB1,smld_symbol_err_prb,'o');
semilogy(SNRindB2,theo_err_prb);
```

m 文件

```
function [pb,ps]=cm_sm32(snr_in_dB)
% [pb,ps]=cm_sm32(snr_in_dB)
%              CM_SM32  finds the probability of bit error and symbol error for the
%              given value of snr_in_dB, signal-to-noise ratio in dB.
N=10000;
E=1;                                   % energy per symbol
snr=10^(snr_in_dB/10);                 % signal-to-noise ratio
sgma=sqrt(E/snr)/2;                    % noise variance
% the signal mapping
s00=[1 0];
s01=[0 1];
s11=[-1 0];
s10=[0 -1];
% generation of the data source
for i=1:N,
  temp=rand;                           % a uniform random variable between 0 and 1
  if (temp<0.25),                      % With probability 1/4, source output is "00."
    dsource1(i)=0;
    dsource2(i)=0;
  elseif (temp<0.5),                   % With probability 1/4, source output is "01."
    dsource1(i)=0;
    dsource2(i)=1;
  elseif (temp<0.75),                  % With probability 1/4, source output is "10."
    dsource1(i)=1;
    dsource2(i)=0;
  else                                 % With probability 1/4, source output is "11."
    dsource1(i)=1;
    dsource2(i)=1;
  end;
end;
% detection and the probability of error calculation
numofsymbolerror=0;
numofbiterror=0;
for i=1:N,
  % The received signal at the detector, for the ith symbol, is:
  n(1)=gngauss(sgma);
  n(2)=gngauss(sgma);
  if ((dsource1(i)==0) & (dsource2(i)==0)),
    r=s00+n;
  elseif ((dsource1(i)==0) & (dsource2(i)==1)),
    r=s01+n;
  elseif ((dsource1(i)==1) & (dsource2(i)==0)),
    r=s10+n;
  else
    r=s11+n;
  end;
  % The correlation metrics are computed below.
  c00=dot(r,s00);
  c01=dot(r,s01);
  c10=dot(r,s10);
```

```
    c11=dot(r,s11);
    % The decision on the ith symbol is made next.
    c_max=max([c00 c01 c10 c11]);
    if (c00==c_max),
      decis1=0; decis2=0;
    elseif (c01==c_max),
      decis1=0; decis2=1;
    elseif (c10==c_max),
      decis1=1; decis2=0;
    else
      decis1=1; decis2=1;
    end;
    % Increment the error counter, if the decision is not correct.
    symbolerror=0;
    if (decis1~=dsource1(i)),
      numofbiterror=numofbiterror+1;
      symbolerror=1;
    end;
    if (decis2~=dsource2(i)),
      numofbiterror=numofbiterror+1;
      symbolerror=1;
    end;
    if (symbolerror==1),
      numofsymbolerror = numofsymbolerror+1;
    end;
end;
ps=numofsymbolerror/N;                % since there are totally N symbols
pb=numofbiterror/(2*N);               % since 2N bits are transmitted
```

7.3.2 差分相位调制与解调

如上所述,相位调制载波信号的解调,需要将载波相位分量 $\psi_1(t)$和 $\psi_2(t)$锁定到接收到的已调载波信号的相位上。一般来说,这意味着接收机在接收信号与两个参考分量 $\psi_1(t)$和 $\psi_2(t)$的互相关中,必须估计出通过信道以后由传输延时产生的接收信号的载波相位偏移,并补偿这载波相位一偏移。通常用一个锁相环路(PLL)来完成载波相位偏移的估计,从而达到相干相位解调的目的。

另一种载波相位调制的类型是差分相位调制,传输数据在调制器之前先进行差分编码。在差分编码中,信息是用相对于前一个信号区间的相移来携带的。例如,在二相相位调制中,信息比特 1 可以用相对于先前载波相位相移 180°相位来传输,而信息比特 0 可以用相对于前一个信号区间的相位相移 0°相位来传输。在四相相位调制中,相继区间之间的相对相移是 0°,90°,180°和 270°,分别对应于信息比特 00,01,11 和 10。对 $M>4$ 的差分编码的推广也是很直接的。用这个编码过程得出的相位调制信号称为**差分编码**。这个编码在调制器之前可用一个相当简单的逻辑电路来实现。

差分编码的相位调制信号的解调和检测可按下述步骤来完成。检测器的接收信号相位 $\theta_r=\arctan(r_2/r_1)$映射到最接近 θ_r 的 M 个可能的传输信号相位$\{\theta_m\}$之一,紧接着这个检测器的是一个相对简单的相位比较器,它在两个相继的信号区间比较这些检测信号的相位,以提取传输的信息。

我们看到,差分编码的相位调制信号的解调不要求载波相位的估计。为了进行详细说明,

假设通过 $r(t)$ 与 $g_T(t)\cos(2\pi f_c t)$ 和 $-g_T(t)\sin(2\pi f_c t)$ 互相关对差分编码信号进行解调。在第 k 个信号区间，解调器输出的两个分量可以用复数形式表示为：

$$r_k = \sqrt{\mathscr{E}_s}\,e^{j(\theta_k - \phi)} + n_k \tag{7.3.20}$$

其中，θ_k 是在第 k 个信号区间上传输的信号的相角，ϕ 是载波相位，$n_k = n_{kc} + jn_{ks}$ 是噪声。类似地，在前一个信号区间上，解调器输出的接收信号向量也是复数量，为：

$$r_{k-1} = \sqrt{\mathscr{E}_s}\,e^{j(\theta_{k-1} - \phi)} + n_{k-1} \tag{7.3.21}$$

相位检测器的判决变量就是这两个复数之间的相位差。等效地说，可以将 r_k 投射到 r_{k-1} 上，并使用所得到的复数相位，即

$$r_k r_{k-1}^* = \mathscr{E}_s e^{j(\theta_k - \theta_{k-1})} + \sqrt{\mathscr{E}_s}\,e^{j(\theta_k - \phi)} n_{k-1} + \sqrt{\mathscr{E}_s}\,e^{-j(\theta_{k-1} - \phi)} n_k + n_k n_{k-1}^* \tag{7.3.22}$$

在无噪声时，这样就得出了相位差 $\theta_k - \theta_{k-1}$。因此，$r_k r_{k-1}^*$ 的均值就与载波相位无关。按照上述方法解调和检测的差分编码 PSK 信号称为差分 PSK（DPSK）。DPSK 的解调和检测如图 7.13 所示。

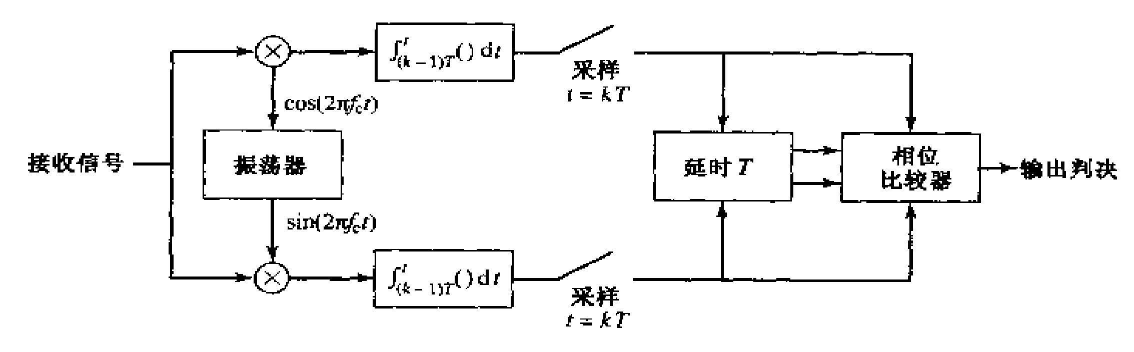


图 7.13　DPSK 解调器方框图

对于二相（$M = 2$）相位调制来说，在 AWGN 信道中导出 DPSK 的差错概率是相当简单的，其结果是：

$$P_2 = \frac{1}{2} e^{-\mathscr{E}_b / N_0} \tag{7.3.23}$$

图 7.14 给出了式(7.3.23)的图，也给出了二相 PSK 的差错概率。由图可见，差错概率低于 10^{-4}，二相 PSK 和二相 DPSK 之间在 SNR 上的差小于 1 dB。

对于 $M > 2$，要准确求得 DPSK 解调器和检测器的差错概率性能是极为困难的。其主要困难在于确定由式(7.3.22)给出的随机变量 $r_k r_{k-1}^*$ 的相位的概率密度函数。然而，一个 DPSK 性能的近似式还是容易得到的，我们来说明这一点。

不失一般性，假定相位差 $\theta_k - \theta_{k-1} = 0$。另外，进一步假设在式(7.3.22)中的指数因子 $e^{-j(\theta_{k-1} - \phi)}$ 和 $e^{j(\theta_k - \phi)}$ 可以吸收进高斯噪声分量 n_{k-1} 和 n_k 中，而不改变它们的统计性质。因此，式(7.3.22)中的 $r_k r_{k-1}^*$ 可以表示成：

$$r_k r_{k-1}^* = \mathscr{E}_s + \sqrt{\mathscr{E}_s}\,(n_k + n_{k-1}^*) + n_k n_{k-1}^* \tag{7.3.24}$$

确定这个相位概率密度函数的复杂性在于 $n_k n_{k-1}^*$ 这一项上。然而，从实际意义上的 SNR 来说，$n_k n_{k-1}^*$ 这一项相对于主要噪声项 $\sqrt{\mathscr{E}_s}\,(n_k + n_{k-1}^*)$ 较小。如果忽略 $n_k n_{k-1}^*$ 这一项，并将 $r_k r_{k-1}^*$ 经由 $\sqrt{\mathscr{E}_s}$ 归一化，那么新的判决标准就变成了：

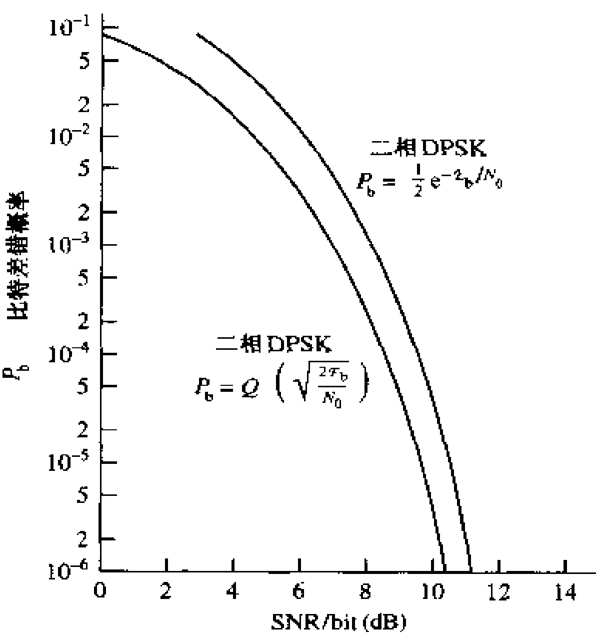


图 7.14 二相 PSK 和 DPSK 的差错概率

$$x = \sqrt{\mathcal{E}_s} + \mathrm{Re}(n_k + n_{k-1}^*)$$

$$y = \mathrm{Im}(n_k + n_{k-1}^*) \tag{7.3.25}$$

变量 x 和 y 是同方差 $\sigma_n^2 = N_0$ 的不相关高斯随机变量。这个相位是:

$$\theta_r = \arctan\frac{y}{x} \tag{7.3.26}$$

这一步所面对的问题与在相位相干解调中所遇到的相同。惟一的差别是现在的噪声方差为 PSK 情况的两倍。据此可得,DPSK 的性能比 PSK 的差 3 dB。这个结果对于 $M \geqslant 4$ 是相当好的,但是对于 $M = 2$ 就不乐观了,因为在大的 SNR 下,二相 DPSK 相对于二相 PSK 来说其损失是小于 3 dB 的。

解说题

解说题 7.4 [DPSK 编码器]

对 $M = 8$ 的 DPSK 实现一个差分编码器。

题解

这个信号点图与 PSK 是相同的,如图 7.8 所示。然而,对于 DPSK 来说,这些信号点代表的是相对于前面的传输信号点相位的相位变化。

实现这个差分编码器的 MATLAB 脚本如下所示。

m文件

```
% MATLAB script for Illustrative Problem 7.4
mapping=[0 1 3 2 7 6 4 5];          % for Gray mapping
M=8;
E=1;
sequence=[0 1 0  0 1 1  0 0 1  1 1 1  1 1 0  0 0 0];
[e]=cm_dpske(E,M,mapping,sequence);      % e is the differential encoder output.
```

解说题

解说题 7.5

完成 $M=4$ 的 DPSK 通信系统的 Monte Carlo 仿真，待仿真的系统的模型如图 7.15 所示。

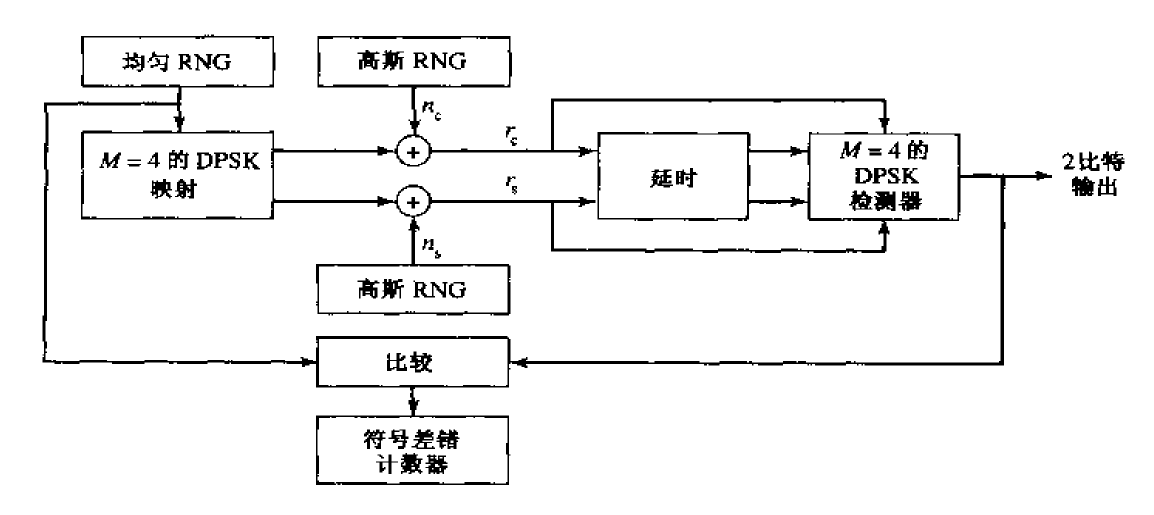


图 7.15　用于 Monte Carlo 仿真的 $M=4$ 的 DPSK 系统方框图

题　解

与在解说题 7.3 中所述的相同，用均匀随机数发生器(RNG)产生{00,01,11,10}比特对，用差分编码将每 2 比特符号映射到 4 种信号点 $s_m=[\cos(\pi m/2),\sin(\pi m/2)]$，$m=0,1,2,3$ 之一。用两个高斯 RNG 产生噪声分量$[n_c,n_s]$，那么接收到的信号加噪声向量是：

$$\begin{aligned}\boldsymbol{r}&=\left[\cos\frac{\pi m}{2}+n_c,\sin\frac{\pi m}{2}+n_s\right]\\&=[r_c,r_s]\end{aligned}$$

差分检测基本上就是将 $\boldsymbol{r}_k$ 和 $\boldsymbol{r}_{k-1}$ 之间的差计算出来。从数学角度上讲，这个计算可以按式(7.3.22)的方式进行，即

$$\begin{aligned}r_kr_{k-1}^*&=(r_{ck}+\mathrm{j}r_{sk})(r_{ck-1}-\mathrm{j}r_{sk-1})\\&=r_{ck}r_{ck-1}+r_{sk}r_{sk-1}+\mathrm{j}(r_{sk}r_{ck-1}-r_{ck}r_{sk-1})\\&=x_k+\mathrm{j}y_k\end{aligned}$$

$\theta_k=\arctan(y_k/x_k)$是相位差。$\theta_k$ 值与可能的相位差{0°,90°,180°,270°}进行比较，并以最接近 θ_k 的相位做出判决。然后将检测出的相位映射到信息比特对。差错计数器对检测序列中的符号差错进行计数。

图 7.16 给出了在不同的 SNR 参数$\mathcal{E}_b/N_0$ 值下，传输 $N=10\ 000$ 个符号时的 Monte Carlo 仿真结果，其中$\mathcal{E}_b=\mathcal{E}_s/2$ 是比特能量。图中还给出了略去 $n_kn_{k-1}^*$项之后近似的符号差错理论值。由图 7.16 可见，近似结果是差错概率的上限。

这个 Monte Carlo 仿真的 MATLAB 脚本如下所示。

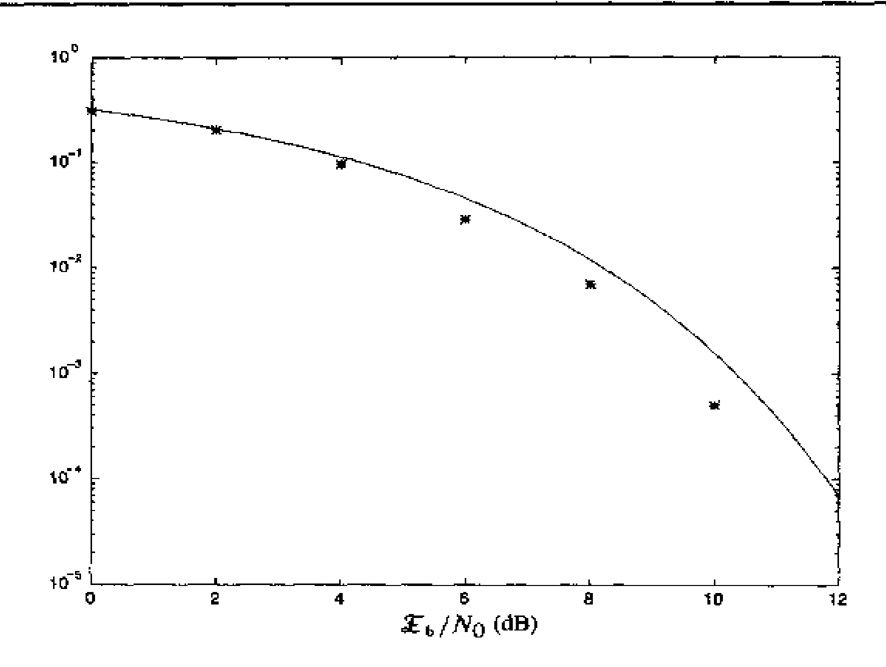


图 7.16　$M=4$ 的 DPSK 系统的 Monte Carlo 仿真性能(实线是基于略去噪声项 $n_k n_{k-1}^*$ 的近似的一个上限)

m 文件

```
% MATLAB script for Illustrative Problem 7.5.
echo on
SNRindB1=0:2:12;
SNRindB2=0:0.1:12;
for i=1:length(SNRindB1),
  smld_err_prb(i)=cm_sm34(SNRindB1(i));   % simulated error rate
  echo off ;
end;
echo on ;
for i=1:length(SNRindB2),
  SNR=exp(SNRindB2(i)*log(10)/10);        % signal-to-noise ratio
  theo_err_prb(i)=2*Qfunct(sqrt(SNR));    % theoretical symbol error rate
  echo off ;
end;
echo on ;
% Plotting commands follow
semilogy(SNRindB1,smld_err_prb,'*');
hold
semilogy(SNRindB2,theo_err_prb);
```

m 文件

```
function [p]=cm_sm34(snr_in_dB)
% [p]=cm_sm34(snr_in_dB)
%         CM_SM34  finds the probability of error for the given
%         value of snr_in_dB, signal-to-noise ratio in dB.
N=10000;
```

```
E=1;                                    % energy per symbol
snr=10^(snr_in_dB/10);                  % signal-to-noise ratio
sgma=sqrt(E/(4*snr));                   % noise variance
% Generation of the data source follows.
for i=1:2*N,
  temp=rand;                            % a uniform random variable between 0 and 1
  if (temp<0.5),
    dsource(i)=0;                       % With probability 1/2, source output is "0."
  else.
    dsource(i)=1;                       % With probability 1/2, source output is "1."
  end;
end;
% Differential encoding of the data source follows.
mapping=[0 1 3 2];
M=4;
[diff_enc_output] = cm_dpske(E,M,mapping,dsource);
% Received signal is then
for i=1:N,
  [n(1) n(2)]=gngauss(sgma);
  r(i,:)=diff_enc_output(i,:)+n;
end;
% detection and the probability of error calculation
numoferr=0;
prev_theta=0;
for i=1:N,
  theta=angle(r(i,1)+j*r(i,2));
  delta_theta=mod(theta-prev_theta,2*pi);
  if ((delta_theta<pi/4) | (delta_theta>7*pi/4)),
    decis=[0 0];
  elseif (delta_theta<3*pi/4),
    decis=[0 1];
  elseif (delta_theta<5*pi/4)
    decis=[1 1];
  else
    decis=[1 0];
  end;
  prev_theta=theta;
  % Increase the error counter, if the decision is not correct.
  if ((decis(1)~=dsource(2*i-1)) | (decis(2)~=dsource(2*i))),
    numoferr=numoferr+1;
  end;
end;
p=numoferr/N;
```

m 文件

```
function [enc_comp] = cm_dpske(E,M,mapping,sequence);
% [enc_comp] = cm_dpske(E,M,mapping,sequence)
%               CM_DPSKE differentially encodes a sequence.
%               E is the average energy, M is the number of constellation points,
%               and mapping is the vector defining how the constellation points are
%               allocated. Finally, "sequence" is the uncoded binary data sequence.
k=log2(M);
N=length(sequence);
```

```
% If N is not divisible by k, append zeros, so that it is...
remainder=rem(N,k);
if (remainder~=0),
  for i=N+1:N+k-remainder,
    sequence(i)=0;
  end;
  N=N+k-remainder;
end;
theta=0;                          % Initially, assume that theta=0.
for i=1:k:N,
  index=0;
  for j=i:i+k-1,
    index=2*index+sequence(j);
  end;
  index=index+1;
  theta=mod(2*pi*mapping(index)/M+theta,2*pi);
  enc_comp((i+k-1)/k,1)=sqrt(E)*cos(theta);
  enc_comp((i+k-1)/k,2)=sqrt(E)*sin(theta);
end;
```

7.4 正交幅度调制

正交幅度调制(QAM)信号使用两个正交载波 $\cos(2\pi f_c t)$ 和 $\sin(2\pi f_c t)$，其中每个都被一个独立的信息比特序列所调制。这个传输信号波形具有如下形式：

$$u_m(t)=A_{mc}g_T(t)\cos(2\pi f_c t)+A_{ms}g_T(t)\sin(2\pi f_c t)\quad m=1,2,\cdots,M \tag{7.4.1}$$

其中，$\{A_{mc}\}$ 和 $\{A_{ms}\}$ 是一组幅度电平，是通过将 k 比特序列映射到信号的幅度而得到的。例如，图 7.17 给出了一个 16 QAM 的信号星座图，它是通过以 $M=4$ 的 PAM 用幅度调制每个正交载波而得到的。一般来说，当两个正交载波中的每个都用 PAM 调制，就会形成矩形的信号星座图。

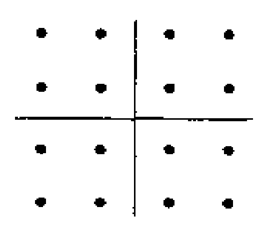


图 7.17　$M=16$ 的 QAM 的信号星座图

更一般地说，QAM 可以看成是一种兼有数字幅度和数字相位调制的形式，因此传输的 QAM 信号波形可以表示成：

$$u_{mn}(t)=A_m g_T(t)\cos(2\pi f_c t+\theta_n),$$
$$m=1,2,\cdots,M_1,n=1,2,\cdots,M_2 \tag{7.4.2}$$

如果 $M_1=2^{k_1}$ 且 $M_2=2^{k_2}$，那么这种兼有幅度和相位调制的方法就形成了符号率为 $R_b/(k_1+k_2)$ 的 $k_1+k_2=\log_2(M_1M_2)$ 个二进制数字的同时传输。图 7.18 是一个 QAM 调制器的功能方框图。

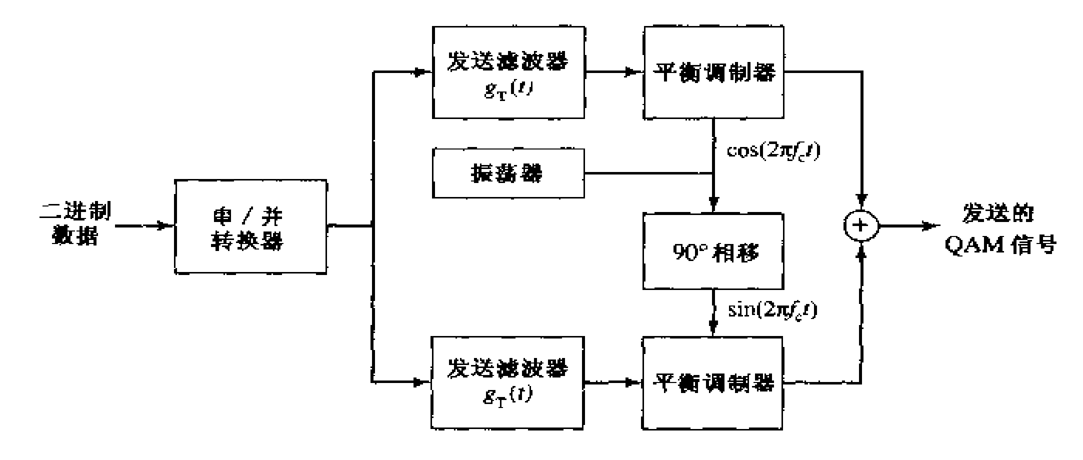


图 7.18　QAM 调制器的功能方框图

很显然,式(7.4.1)和式(7.4.2)给出的信号的几何表示利用了二维信号向量形式:

$$s_m = (\sqrt{\mathscr{E}_s}A_{mc} \quad \sqrt{\mathscr{E}_s}A_{ms}),\quad m = 1,2,\cdots,M \tag{7.4.3}$$

图 7.19 给出了 QAM 信号空间星座图的几个例子。注意,$M = 4$ 的 QAM 与 $M = 4$ 的 PSK,其星座图是一样的。

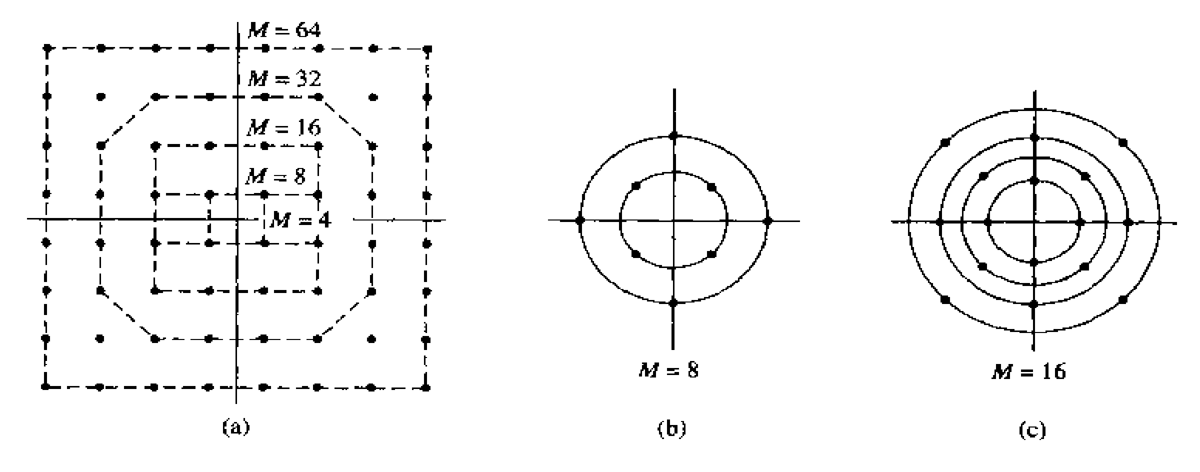


图 7.19　(a)是矩形星座图,(b)和(c)是圆形 QAM 信号的星座图

7.4.1　QAM 的解调和检测

现在,假设信号在通过信道的传输过程中引入了载波相位偏移。另外,接收信号受到加性高斯噪声的污损。因此,$r(t)$可以表示为:

$$r(t) = A_{mc}g_T(t)\cos(2\pi f_c t + \phi) + A_{ms}g_T(t)\sin(2\pi f_c t + \phi) + n(t) \tag{7.4.4}$$

其中,ϕ 是载波相位偏移,而

$$n(t) = n_c(t)\cos(2\pi f_c t) - n_s(t)\sin(2\pi f_c t)$$

将接收信号与两个相移的基函数:

$$\psi_1(t) = g_T(t)\cos(2\pi f_c t + \phi)$$

$$\psi_2(t) = g_T(t)\sin(2\pi f_c t + \phi) \tag{7.4.5}$$

做相关,如图 7.20 所示,而且将相关器的输出采样并送至检测器。图 7.20 中给出的锁相

环(PLL)估计出接收信号的载波相位偏移 ϕ,并通过式(7.4.5)所示的 $\psi_1(t)$和 $\psi_2(t)$的相移来补偿这个相位偏移。假定图 7.20 中的时钟已与接收信号同步,使得相关器的输出在合适的瞬间采样。在这些条件下,两个相关器的输出是:

$$
\begin{aligned}
r_c &= A_{mc} + n_c\cos\phi - n_s\sin\phi \\
r_s &= A_{ms} + n_c\sin\phi + n_s\cos\phi
\end{aligned} \tag{7.4.6}
$$

其中,n_c 和 n_s 分别是:

$$
\begin{aligned}
n_c &= \frac{1}{2}\int_0^T n_c(t)g_T(t)\mathrm{d}t \\
n_s &= \frac{1}{2}\int_0^T n_s(t)g_T(t)\mathrm{d}t
\end{aligned} \tag{7.4.7}
$$

这些都是零均值、方差为 $\sigma^2 = N_0/2$ 的不相关高斯随机变量。

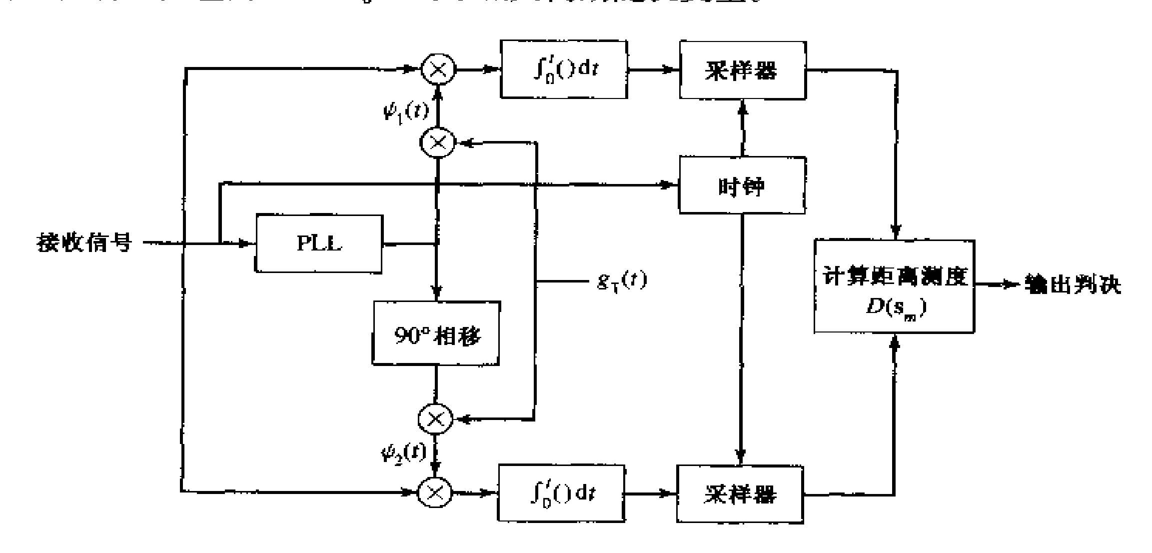


图 7.20 QAM 信号的解调与检测

最佳检测器计算了距离测度:

$$D(\boldsymbol{r},\boldsymbol{s}_m) = |\boldsymbol{r} - \boldsymbol{s}_m|^2, \quad m = 1,2,\cdots,M \tag{7.4.8}$$

其中,$\boldsymbol{r}' = (r_1, r_2)$,$\boldsymbol{s}_m$ 由式(7.4.3)给出。

7.4.2 AWGN 信道中的 QAM 的差错概率

这一节要考虑采用矩形信号星座图的 QAM 系统的性能。矩形 QAM 信号星座有个突出的优点:很容易按照在相位正交载波上所加的两个 PAM 信号来产生。另外,它们也容易被解调出来。

在矩形信号星座图中 $M = 2^k$,其中 k 是偶数,这个 QAM 信号星座图等效于在正交载波上的两个 PAM 信号,其中每个都有$\sqrt{M} = 2^{k/2}$个信号点。由于相位正交的信号分量用相干检测可以完全分开,所以 QAM 的差错概率很容易由 PAM 的差错概率确定。具体地说,对于 M 电平 QAM 系统,一个正确判决的概率是:

$$P_c = (1 - P_{\sqrt{M}})^2 \tag{7.4.9}$$

其中,$P_{\sqrt{M}}$是在这个等效 QAM 系统的每个正交信号中具有一半平均功率的$\sqrt{M}$电平 PAM 系统

的差错概率。通过对 PAM 差错概率进行适当的修正，可得：

$$P_{\sqrt{M}} = 2\left(1 - \frac{1}{\sqrt{M}}\right)Q\left(\sqrt{\frac{3}{M-1}\frac{\mathcal{E}_{av}}{N_0}}\right) \tag{7.4.10}$$

其中，$\mathcal{E}_{av}/N_0$ 是每个符号的平均 SNR。因此，对于 M 电平 QAM 系统，一个符号差错的概率是：

$$P_M = 1 - (1 - P_{\sqrt{M}})^2 \tag{7.4.11}$$

值得一提的是，这个结果对于 $M = 2^k$，k 为偶数时是精确的。另外，当 k 为奇数时，不存在等效的 $\sqrt{M}$ 电平 PAM 系统。然而，这也不会有什么问题，因为对一个矩形信号集来说很容易确定误码率。如果采用基于式(7.4.8)给出的最佳距离测度作为判据的最佳检测器，那么可以相当直接地证明符号差错概率是严格地以下式为上界的：

$$P_M \leqslant 1 - \left[1 - 2Q\left(\sqrt{\frac{3\mathcal{E}_{av}}{(M-1)N_0}}\right)\right]^2$$

$$\leqslant 4Q\left(\sqrt{\frac{3k\,\mathcal{E}_{avb}}{(M-1)N_0}}\right) \tag{7.4.12}$$

该式对任何 $k \geqslant 1$ 成立，其中 $\mathcal{E}_{avb}/N_0$ 是每比特的平均 SNR。作为每比特平均 SNR 的函数，一个符号差错的概率如图 7.21 所示。

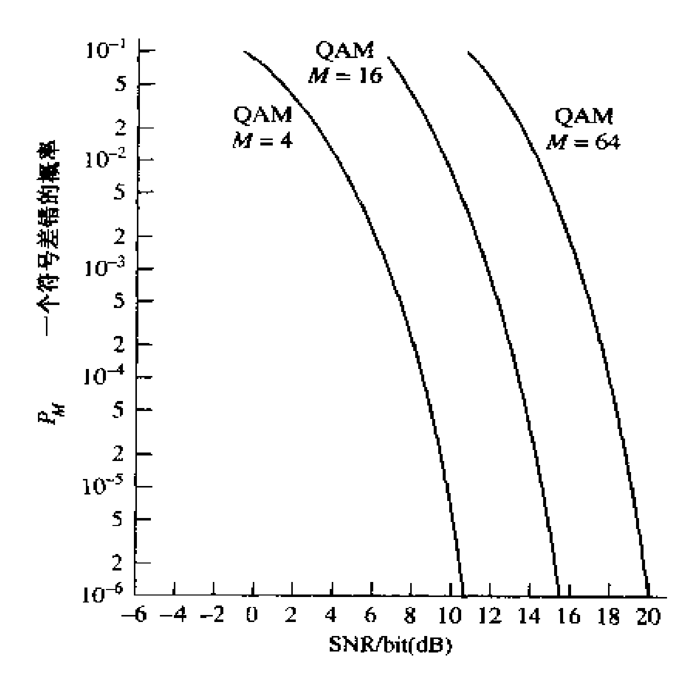


图 7.21　QAM 中一个符号差错的概率

解说题

解说题 7.6　[QAM 仿真]

完成一个利用矩形信号星座图的 $M = 16$ 的 QAM 通信系统的 Monte Carlo 仿真。待仿真的系统的模型如图 7.22 所示。

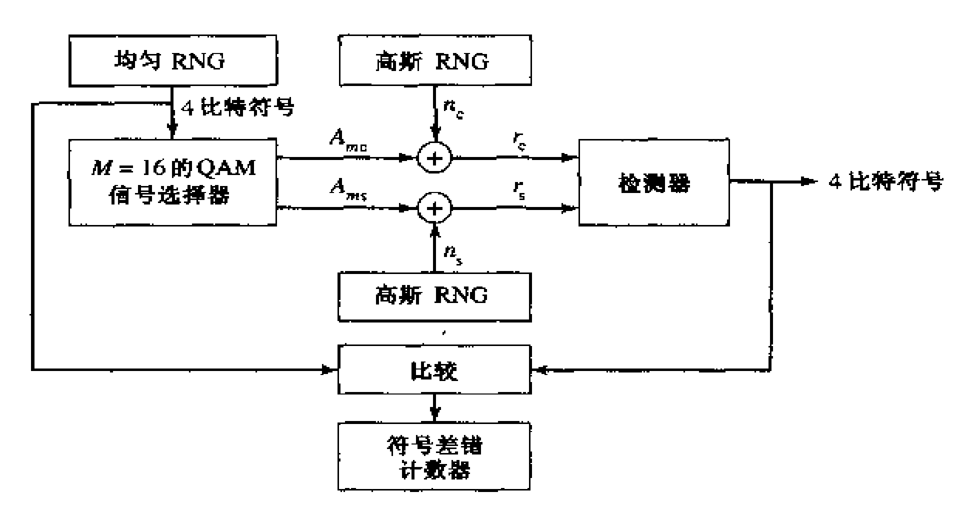


图 7.22　$M = 16$ 的 QAM 系统 Monte Carlo 仿真的方框图

[illegible]

用均匀随机数发生器产生对应于 16 种可能的由 b_1, b_2, b_3, b_4 组成的 4 比特信息符号序列。将这个信息符号序列映射到对应的信号点,如图 7.23 所示,其坐标为$[A_{mc}, A_{ms}]$。用两个高斯随机数发生器产生噪声分量$[n_c, n_s]$。为简单起见,我们把信道相移 ϕ 置为 0。这样,接收到的信号加噪声向量是:

$$\boldsymbol{r} = [A_{mc} + n_c \quad A_{ms} + n_s]$$

检测器计算由式(7.4.8)给出的距离测度,并用最接近接收信号向量 $\boldsymbol{r}$ 的信号点做出判定。差错计数器对检测序列中的符号差错进行计数。图 7.24 给出的是在不同的 SNR 参数$\mathscr{E}_b/N_0$ 值下,传输 $N = 10\ 000$ 个符号的 Monte Carlo 仿真结果,这里$\mathscr{E}_b = \mathscr{E}_s/4$ 是比特能量。图中也给出了由式(7.4.10)和式(7.4.11)给出的符号差错概率的理论值。

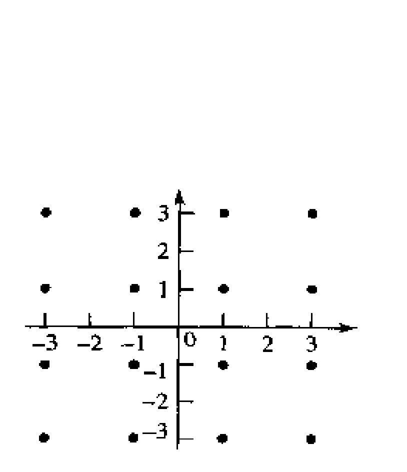


图 7.23　$M = 16$ 的 QAM 系统 Monte Carlo 仿真信号星座图

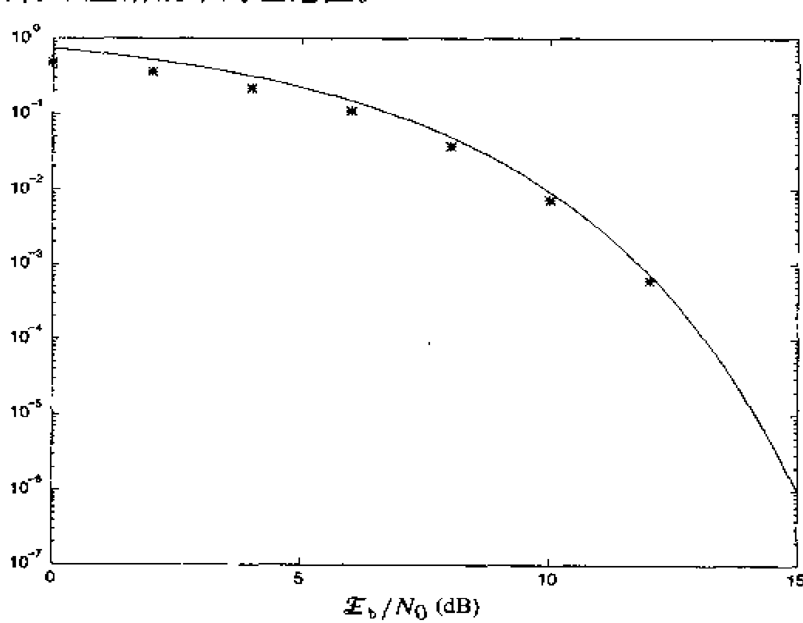


图 7.24　$M = 16$ 的 QAM 系统 Monte Carlo 仿真性能

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 7.6
echo on
SNRindB1=0:2:15;
SNRindB2=0:0.1:15;
M=16;
k=log2(M);
for i=1:length(SNRindB1),
  smld_err_prb(i)=cm_sm41(SNRindB1(i));   % simulated error rate
  echo off;
end;
echo on ;
for i=1:length(SNRindB2),
  SNR=exp(SNRindB2(i)*log(10)/10);         % signal-to-noise ratio
  % theoretical symbol error rate
  theo_err_prb(i)=4*Qfunct(sqrt(3*k*SNR/(M-1)));
  echo off ;
end;
echo on ;
% Plotting commands follow.
semilogy(SNRindB1,smld_err_prb,'*');
hold
semilogy(SNRindB2,theo_err_prb);
```

m 文件

```
function [p]=cm_sm41(snr_in_dB)
% [p]=cm_sm41(snr_in_dB)
%              CM_SM41  finds the probability of error for the given
%              value of snr_in_dB, SNR in dB.
N=10000;
d=1;                                % min. distance between symbols
Eav=10*d^2;                         % energy per symbol
snr=10^(snr_in_dB/10);              % SNR per bit (given)
sgma=sqrt(Eav/(8*snr));             % noise variance
M=16;
% Generation of the data source follows.
for i=1:N,
  temp=rand;                        % a uniform R.V. between 0 and 1
  dsource(i)=1+floor(M*temp);       % a number between 1 and 16, uniform
end;
% Mapping to the signal constellation follows.
mapping=[-3*d 3*d;
            -d  3*d;
             d  3*d;
           3*d  3*d;
          -3*d   d;
            -d   d;
             d   d;
           3*d   d;
```

```
        -3*d  -d;
          -d  -d;
           d  -d;
         3*d  -d;
        -3*d  -3*d;
          -d  -3*d;
           d  -3*d;
         3*d  -3*d];
for i=1:N,
  qam_sig(i,:)=mapping(dsource(i),:);
end;
% received signal
for i=1:N,
  [n(1) n(2)]=gngauss(sgma);
  r(i,:)=qam_sig(i,:)+n;
end;
% detection and error probability calculation
numoferr=0;
for i=1:N,
  % Metric computation follows.
  for j=1:M,
    metrics(j)=(r(i,1)-mapping(j,1))^2+(r(i,2)-mapping(j,2))^2;
  end;
  [min_metric decis] = min(metrics);
  if (decis~=dsource(i)),
    numoferr=numoferr+1;
  end;
end;
p=numoferr/(N);
```

7.5 载波频率调制

我们已经讨论过通过调制载波幅度、载波相位,或者兼有幅度和相位的几种传输数字信息的方法。数字信息也能够通过调制载波的频率进行传输。

从下面的讨论中可以看到,用频率调制进行数字传输是一种适合于缺乏相位稳定性的信道的调制方法,而相位稳定性对实现载波相位估计来说是必不可少的。我们已经介绍过的几种线性调制方法,如PAM,相干PSK以及QAM,都需要载波相位估计以实现相位相干检测。

7.5.1 频移键控

频率调制的最简单形式是二进制频移键控(frequency-shift keying,简称FSK)。在二进制FSK中使用了两个不同的频率,例如 f_1 和 $f_2=f_1+\Delta f$ 来传输一个二进制的信息序列。稍后我们再考虑频率间隔 $\Delta f=f_2-f_1$ 的选取问题。因此,这两个信号波形可以表示为:

$$u_1(t)=\sqrt{\frac{2\mathscr{E}_b}{T_b}}\cos(2\pi f_1 t),\quad 0\leqslant t\leqslant T_b$$

$$u_2(t)=\sqrt{\frac{2\mathscr{E}_b}{T_b}}\cos(2\pi f_2 t),\quad 0\leqslant t\leqslant T_b \tag{7.5.1}$$

其中,$\mathscr{E}_b$ 是每比特信号能量。T_b 是比特区间的持续期。

更一般地，可以用 M FSK 以每个信号波形传输一组 $k=\log_2 M$ 个比特信息。这时，这 M 个信号波形可以表示为：

$$u_m(t)=\sqrt{\frac{2\mathcal{E}_s}{T}}\cos(2\pi f_c t+2\pi m\Delta f t),$$
$$m=0,1,\cdots,M-1,0\leqslant t\leqslant T \tag{7.5.2}$$

其中，$\mathcal{E}_s=k\mathcal{E}_b$ 是每个符号的能量，$T=kT_b$ 是符号区间，而 Δf 是相继两个频率之间的频率间隔，即 $\Delta f=f_m-f_{m-1}$，$m=1,2,\cdots,M-1$，而 $f_m=f_c+m\Delta f$。

注意，M FSK 的波形具有相等的能量 $\mathcal{E}_s$。频率间隔 Δf 决定了能够在这 M 个可能传输信号之间的鉴别程度。作为一对信号波形之间的相似性（或非相似性）的一种度量，利用相关系数 γ_{mn}：

$$\gamma_{mn}=\frac{1}{\mathcal{E}_s}\int_0^T u_m(t)u_n(t)\mathrm{d}t \tag{7.5.3}$$

将 $u_m(t)$ 和 $u_n(t)$ 代入式(7.5.3)，可得

$$\begin{aligned}\gamma_{mn}&=\frac{1}{\mathcal{E}_s}\int_0^T\frac{2\mathcal{E}_s}{T}\cos(2\pi f_c t+2\pi m\Delta f t)\cos(2\pi f_c t+2\pi n\Delta f t)\mathrm{d}t\\&=\frac{1}{T}\int_0^T\cos2\pi(m-n)\Delta f t\,\mathrm{d}t+\frac{1}{T}\int_0^T\cos[4\pi f_c t+2\pi(m+n)\Delta f t]\mathrm{d}t\\&=\frac{\sin[2\pi(m-n)\Delta f T]}{2\pi(m-n)\Delta f T}\end{aligned} \tag{7.5.4}$$

其中，当 $f_c\gg 1/T$ 时第 2 个积分为零。γ_{mn} 作为频率间隔 Δf 的函数，其图形如图 7.25 所示。由图可见，当 Δf 是 $1/(2T)$ 的倍数时，这些信号波形是正交的。所以，对于正交性来说，相继频率之间的最小频率间隔是 $1/(2T)$。同时还要注意到，相关系数的最小值是 $\gamma_{mn}=-0.217$，它发生在频率间隔 $\Delta f=0.715/T$ 处。

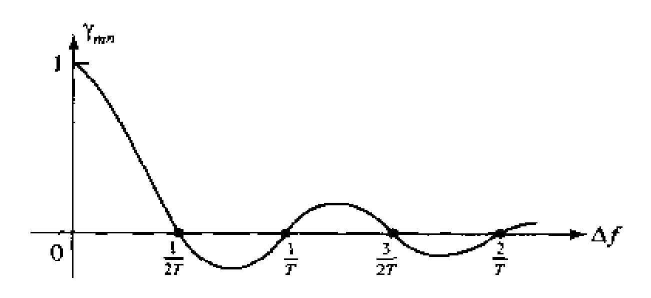


图 7.25　作为频率间隔的函数的 FSK 信号的互相关系数

M 个正交 FSK 波形在几何上可用 M 个 M 维正交向量表示如下：

$$\boldsymbol{s}_0=(\sqrt{\mathcal{E}_s},0,\cdots,0) \tag{7.5.5}$$
$$\boldsymbol{s}_1=(0,\sqrt{\mathcal{E}_s},0,\cdots,0) \tag{7.5.6}$$
$$\vdots \tag{7.5.7}$$
$$\boldsymbol{s}_{M-1}=(0,0,\cdots,0,\sqrt{\mathcal{E}_s}) \tag{7.5.8}$$

其中，基函数是 $\psi_m(t)=\sqrt{2/T}\cos[2\pi(f_c+m\Delta f)t]$。各对信号向量之间的距离是 $d=\sqrt{2\mathcal{E}_s}$（对全部 m,n），它也是 M 个信号之间的最小距离。应该注意到，这些信号与 5.4 节讨过的 M

个基带正交信号是等效的。

下面考虑 M FSK 信号的解调和检测。

7.5.2 FSK 信号的解调和检测

假定 FSK 信号是经由加性白高斯噪声信道传输的,并假设每个信号在通过信道传输时都产生了延时,这样在解调器输入端的这个滤波后的接收信号可以表示为:

$$r(t)=\sqrt{\frac{2\mathcal{E}_s}{T}}\cos(2\pi f_c t+2\pi m\Delta f t+\phi_m)+n(t) \tag{7.5.9}$$

其中,ϕ_m 代表第 m 个信号由于传输延时而产生的相移,$n(t)$代表加性带通噪声,可以表示为:

$$n(t)=n_c(t)\cos(2\pi f_c t)-n_s(t)\sin(2\pi f_c t) \tag{7.5.10}$$

可以用两种方法完成这个 M FSK 信号的解调和检测。一种方法是要估计出这 M 个载波相移$\{\phi_m\}$,并进行**相位相干解调和检测**。作为一种替代方法,可以在解调和检测中不考虑载波相位。

在相位相干解调中,将接收信号 $r(t)$与每 M 个可能接收的信号 $\cos(2\pi f_c t+2\pi m\Delta f t+\hat{\phi}_m)$, $m=0,1,\cdots,M-1$ 做相关,其中$\{\hat{\phi}_m\}$是载波相位的估值。图 7.26 给出了这种解调类型的一个方框图。有意思的是,当 $\hat{\phi}_m\neq\phi_m$(不准确的相位估值)时,$m=0,1,\cdots,M-1$,在解调器端信号正交性所需的频率间隔是 $\Delta f=1/T$,它是在 $\phi=\hat{\phi}$时正交性所需的最小间隔的两倍。

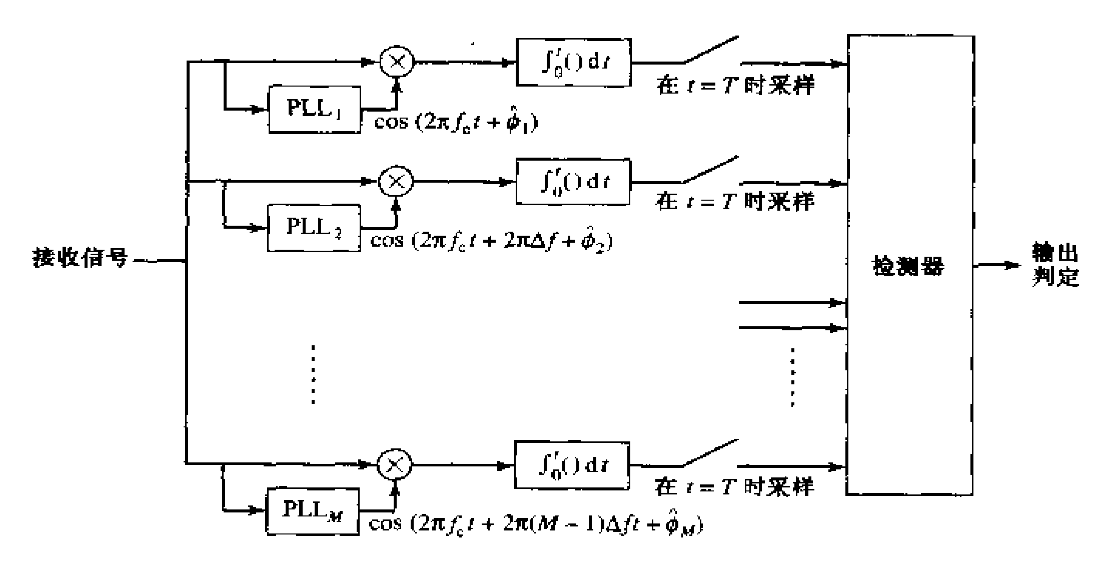


图 7.26 M FSK 信号的相位相干解调

估计 M 个载波相位的需求使得 FSK 信号的相干解调变得极为复杂和不切实际,当信号数目很多时尤其如此。因此,我们将不考虑 FSK 信号的相干检测。

现在来考虑一种不要求这些载波相位知识的解调和检测方法。这种解调可按图 7.27 所示的原理来完成。在这种情况下,每个信号波形有两个相关器,或者说,一般总共有 $2M$ 个相关器。将接收信号与基函数(正交载波)$\sqrt{2/T}\cos(2\pi f_c t+2\pi m\Delta f t)$和 $\sqrt{2/T}\sin(2\pi f_c t+2\pi m\Delta f t)$, $m=0,1,\cdots,M-1$ 做相关。这些相关器的 $2M$ 个输出在该信号区间的末端被采样并送至检测器。于是,如果传输的是第 m 个信号,那么在检测器输入的 $2M$ 个样本可以表示为:

$$r_{kc}=\sqrt{\mathcal{E}_s}\left[\frac{\sin[2\pi(k-m)\Delta fT]}{2\pi(k-m)\Delta fT}\cos\phi_m-\frac{\cos[2\pi(k-m)\Delta fT]-1}{2\pi(k-m)\Delta fT}\sin\phi_m\right]+n_{kc}$$

$$r_{ks}=\sqrt{\mathcal{E}_s}\left[\frac{\cos[2\pi(k-m)\Delta fT]-1}{2\pi(k-m)\Delta fT}\cos\phi_m+\frac{\sin[2\pi(k-m)\Delta fT]}{2\pi(k-m)\Delta fT}\sin\phi_m\right]+n_{ks}\quad(7.5.11)$$

其中，n_{kc}和 n_{ks}代表在采样输出中的高斯噪声分量。

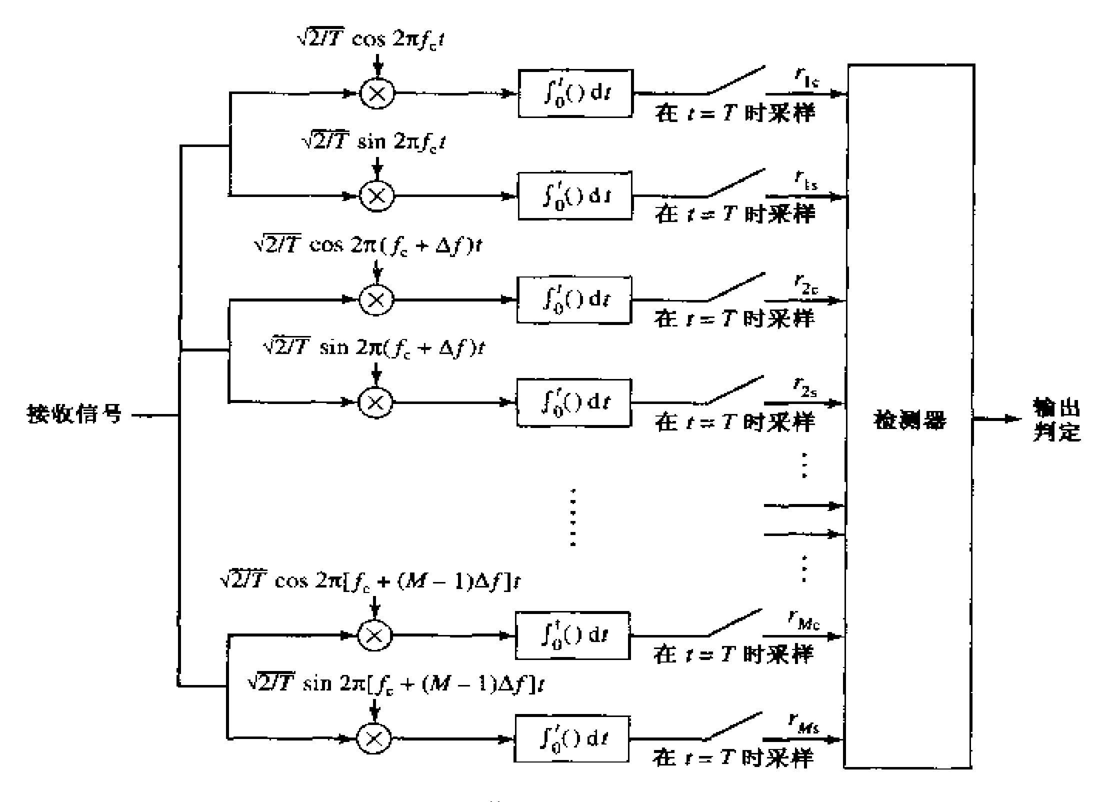


图 7.27　M 信号的非相干检测解调

可以看到，当 $k=m$ 时，对检测器的采样值是：

$$r_{mc}=\sqrt{\mathcal{E}_s}\cos\phi_m+n_{mc}$$

$$r_{ms}=\sqrt{\mathcal{E}_s}\sin\phi_m+n_{ms}\quad(7.5.12)$$

我们还看到 $k\neq m$ 时在样本 r_{kc}和 r_{ks}中的信号分量将是零，只要相继频率之间的频率间隔是 $\Delta f=1/T$，就与相移 ϕ_k 的值无关。在这样的情况下，其余的 $2(M-1)$个相关器的输出仅由噪声组成，即

$$r_{kc}=n_{kc},\ r_{ks}=n_{ks},\ k\neq m\quad(7.5.13)$$

在下面的推导中都假设 $\Delta f=1/T$，以保证信号是正交的。

可以证明，这 $2M$ 个噪声样本$\{n_{kc}\}$和$\{n_{ks}\}$都是零均值，具有相等方差 $\sigma^2=N_0/2$ 且互不相关的高斯随机变量。这样，在条件 ϕ_m 下 r_{mc}和 r_{ms}的联合概率密度函数是：

$$f_{r_m}(r_{mc},r_{ms}\mid\phi_m)=\frac{1}{2\pi\sigma^2}e^{-[(r_{mc}-\sqrt{\mathcal{E}_s}\cos\phi_m)^2+(r_{ms}-\sqrt{\mathcal{E}_s}\sin\phi_m)^2]/2\sigma^2}\quad(7.5.14)$$

并且，对 $m\neq k$ 有：

$$f_{r_k}(r_{kc},r_{ks})=\frac{1}{2\pi\sigma^2}e^{-(r_{kc}^2+r_{ks}^2)/(2\sigma^2)}\quad(7.5.15)$$

已知 $2M$ 个观察到的随机变量$\{r_{kc}, r_{ks}\}_{k=0}^{M-1}$，最佳检测器将对应于后验概率最大的信号选出，即

$$P[s_m \text{ 被传输} \mid \boldsymbol{r}] \equiv P(s_m \mid \boldsymbol{r}), \quad m = 0, 1, \cdots, M-1 \tag{7.5.16}$$

其中，$\boldsymbol{r}$ 是元素为$\{r_{kc}, r_{ks}\}_{k=0}^{M-1}$的 $2M$ 维向量。当这些信号是等概率的时，由式(7.5.16)给出的最佳检测器计算信号包络，定义为：

$$r_m = \sqrt{r_{mc}^2 + r_{ms}^2}, \quad m = 0, 1, \cdots, M-1 \tag{7.5.17}$$

并选出对应于集合$\{r_m\}$的最大包络的信号。这种情况下的最佳检测器称为**包络检测器**。

一种等效的检测器是计算平方包络：

$$r_m^2 = r_{mc}^2 + r_{ms}^2, \quad m = 0, 1, \cdots, M-1 \tag{7.5.18}$$

并选出对应于$\{r_m^2\}$的最大值的信号。这种情况下的最佳检测器称为**平方律检测器**。

解说题 7.7 [FSK 信号]

考虑一个二进制通信系统，它采用由下式给出的 2 FSK 信号波形：

$$u_1(t) = \cos(2\pi f_1 t), \quad 0 \leqslant t \leqslant T_b$$

$$u_2(t) = \cos(2\pi f_2 t), \quad 0 \leqslant t \leqslant T_b$$

其中，$f_1 = 1000/T_b$，$f_2 = f_1 + 1/T_b$。信道给予每个传输信号的相移为 $\phi = 45°$，因此在无噪声情况下接收信号是：

$$r(t) = \cos\left(2\pi f_i t + \frac{\pi}{4}\right), \quad i = 1, 2, \quad 0 \leqslant t \leqslant T_b$$

用数值方法实现该 FSK 信号的相关型解调器。

在比特区间 T_b 中以 $F_s = 5000/T_b$ 的采样率对接收信号 $r(t)$采样，这样该接收信号 $r(t)$用 5000样本$\{r(n/F_s)\}$表示。如图 7.27 所示，相关解调器将$\{r(n/F_s)\}$乘以 $u_1(t) = \cos(2\pi f_1 t)$，$v_1(t) = \sin(2\pi f_1 t)$，$u_2(t) = \cos(2\pi f_2 t)$，$v_2(t) = \sin(2\pi f_2 t)$的采样，于是这些相关器的输出是：

$$r_{1c}(k) = \sum_{n=0}^{k} r\left(\frac{n}{F_s}\right) u_1\left(\frac{n}{F_s}\right), \quad k = 1, 2, \cdots, 5000$$

$$r_{1s}(k) = \sum_{n=0}^{k} r\left(\frac{n}{F_s}\right) v_1\left(\frac{n}{F_s}\right), \quad k = 1, 2, \cdots, 5000$$

$$r_{2c}(k) = \sum_{n=0}^{k} r\left(\frac{n}{F_s}\right) u_2\left(\frac{n}{F_s}\right), \quad k = 1, 2, \cdots, 5000$$

$$r_{2s}(k) = \sum_{n=0}^{k} r\left(\frac{n}{F_s}\right) v_2\left(\frac{n}{F_s}\right), \quad k = 1, 2, \cdots, 5000$$

检测器是一个平方律检测器，它计算两个判决变量：

$$r_1 = r_{1c}^2(5000) + r_{1s}^2(5000)$$

$$r_2 = r_{2c}^2(5000) + r_{2s}^2(5000)$$

并对应于最大判决变量选出信息比特。

用数值方法实现相关的 MATLAB 程序如下所示。图 7.28 给出的是基于信号 $u_1(t)$传输时相关器输出的图。

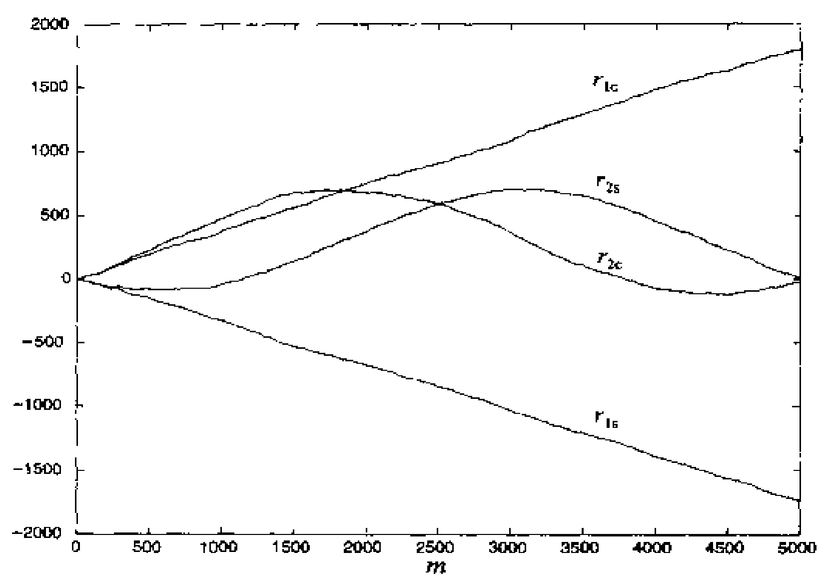


图 7.28　用于二进制 FSK 解调的相关器输出

m 文件

```
% MATLAB script for Illustrative Problem 7.7.
echo on
Tb=1;
f1=1000/Tb;
f2=f1+1/Tb;
phi=pi/4;
N=5000;                              % number of samples
t=0:Tb/(N-1):Tb;
u1=cos(2*pi*f1*t);
u2=cos(2*pi*f2*t);
% Assuming that u1 is transmitted, the received signal r is
sgma=1;                              % noise variance
for i=1:N,
  r(i)=cos(2*pi*f1*t(i)+phi)+gngauss(sgma);
  echo off;
end;
echo on ;
% The correlator outputs are computed next.
v1=sin(2*pi*f1*t);
v2=sin(2*pi*f2*t);
r1c(1)=r(1)*u1(1);
r1s(1)=r(1)*v1(1);
r2c(1)=r(1)*u2(1);
r2s(1)=r(1)*v2(1);
for k=2:N,
  r1c(k)=r1c(k-1)+r(k)*u1(k);
  r1s(k)=r1s(k-1)+r(k)*v1(k);
  r2c(k)=r2c(k-1)+r(k)*u2(k);
  r2s(k)=r2s(k-1)+r(k)*v2(k);
  echo off;
end;
echo on;
% decision variables
r1=r1c(5000)^2+r1s(5000)^2;
r2=r2c(5000)^2+r2s(5000)^2;
% Plotting commands follow.
```

7.5.3 FSK 非相干检测的差错概率

对 M FSK 信号的最佳包络检测器的性能推导，在大多数有关数字通信的教材中都能找到。一个符号差错的概率可以表示为：

$$P_M = \sum_{n=1}^{M-1}(-1)^{n+1}\binom{M-1}{n}\frac{1}{n+1}e^{-nk\mathcal{E}_b/[N_0(n+1)]} \tag{7.5.19}$$

当 $M=2$ 时，这个表达式就变成二进制 FSK 的差错概率：

$$P_2 = \frac{1}{2}e^{-\mathcal{E}_b/(2N_0)} \tag{7.5.20}$$

对于 $M>2$，利用下面的关系：

$$P_b = \frac{2^{k-1}}{2^k-1}P_M \tag{7.5.21}$$

可以从符号差错概率得出比特差错概率。

对于 $M=2,4,8,16$ 和 32，作为每比特 SNR 的函数的比特差错概率如图 7.29 所示。由图可见，对于任意给定的比特差错概率，每比特 SNR 随 M 增加而降低。在 $M\to\infty$ 的极限情况下，只要每比特 SNR 超过 -1.6 dB，差错概率就能达到任意小。这就是对任何经由 AWGN 信道传输信息的数字通信系统的信道容量极限，或称香农(Shannon)极限。

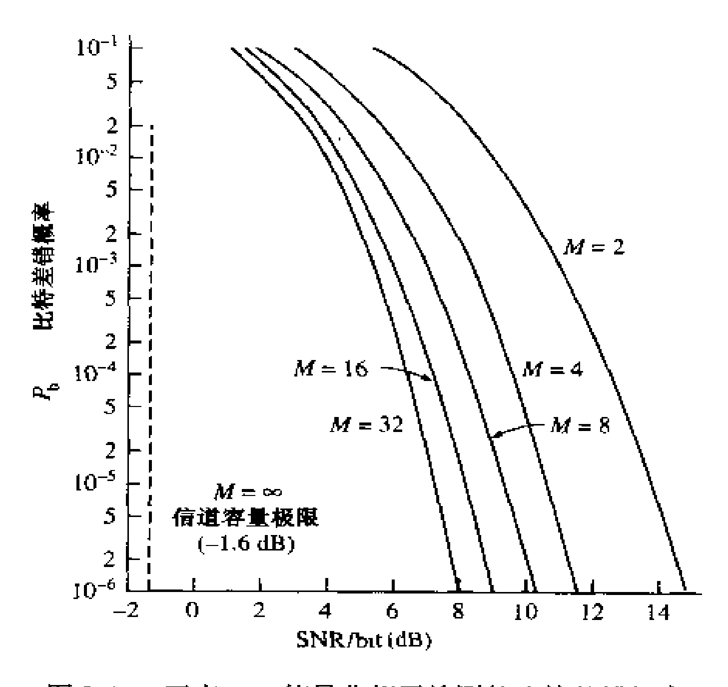


图 7.29　正交 FSK 信号非相干检测的比特差错概率

增加 M 的代价是传输这些信号所需的带宽。因为相邻频率之间的频率间隔是 $\Delta f = 1/T$（对正交信号），对 M 个信号所需的带宽就是 $W = M/T$。比特率是 $R = k/T$，$k = \log_2 M$，因此比特率与带宽的比是：

$$\frac{R}{W} = \frac{\log_2 M}{M} \tag{7.5.22}$$

可见随 $M\to\infty$，$R/W\to 0$。

解说题

解说题 7.8 [二进制 FSK 仿真]

完成一个二进制 FSK 通信系统的 Monte Carlo 仿真，其中信号波形由式(7.5.1)给出，$f_2 = f_1 + 1/T_b$，检测器为平方律检测器。待仿真的二进制 FSK 系统的方框图如图 7.30 所示。

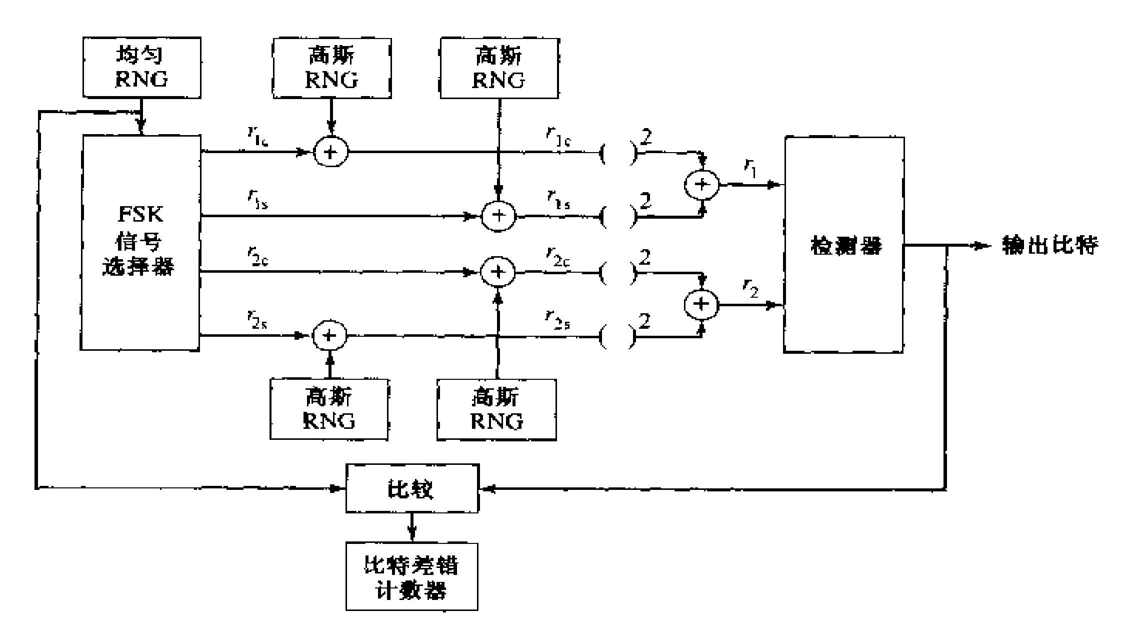


图 7.30 用于 Monte Carlo 仿真的二进制 FSK 系统的方框图

题 解

因为这些信号是正交的，当 $u_1(t)$ 传输时，第一个解调器的输出是：

$$r_{1c} = \sqrt{\mathscr{E}_b}\cos\phi + n_{1c}$$

$$r_{1s} = \sqrt{\mathscr{E}_b}\sin\phi + n_{1s}$$

而第二个解调器的输出是：

$$r_{2c} = n_{2c}$$

$$r_{2s} = n_{2s}$$

其中，n_{1c}，n_{1s}，n_{2c}和 n_{2s}都是互为统计独立的零均值高斯随机变量，方差均为 σ^2，ϕ 代表信道相移。

在上面的表达式中，为简单起见，可以置信道相移 ϕ 为零。平方律检测器计算

$$r_1 = r_{1c}^2 + r_{1s}^2$$

$$r_2 = r_{2c}^2 + r_{2s}^2$$

并选出对应于这两个判决变量中较大的那个的信息比特。差错计数器通过比较传输序列和检测器输出，测出误码率。图 7.31 给出的是测得的误码率，并将它与由式(7.5.20)给出的理论差错概率进行了对比。

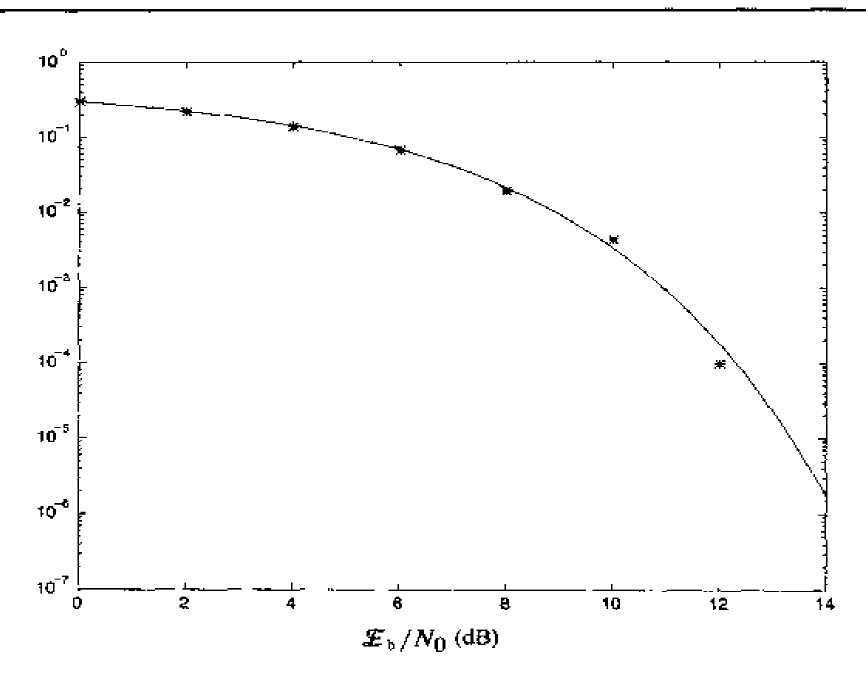


图 7.31 由 Monte Carlo 仿真所得的二进制 FSK 系统的性能

m 文件

```
% MATLAB script for Illustrative Problem 7.8.
echo on
SNRindB1=0:2:15;
SNRindB2=0:0.1:15;
for i=1:length(SNRindB1),
  smld_err_prb(i)=cm_sm52(SNRindB1(i));   % simulated error rate
  echo off ;
end;
echo on ;
for i=1:length(SNRindB2),
  SNR=exp(SNRindB2(i)*log(10)/10);        % signal-to-noise ratio
  theo_err_prb(i)=(1/2)*exp(-SNR/2);      % theoretical symbol error rate
  echo off;
end;
echo on;
% Plotting commands follow.
semilogy(SNRindB1,smld_err_prb,'*');
hold
semilogy(SNRindB2,theo_err_prb);
```

m 文件

```
function [p]=cm_sm52(snr_in_dB)
% [p]=cm_sm52(snr_in_dB)
%               CM_SM52  Returns the probability of error for the given
%               value of snr_in_dB, signal-to-noise ratio in dB.
N=10000;
Eb=1;
d=1;
snr=10^(snr_in_dB/10);                    % signal-to-noise ratio per bit
```

```
sgma=sqrt(Eb/(2*snr));                    % noise variance
phi=0;
% Generation of the data source follows.
for i=1:N,
  temp=rand;                              % a uniform random variable between 0 and 1
  if (temp<0.5),
    dsource(i)=0;
  else
    dsource(i)=1;
  end;
end;
% detection and the probability of error calculation
numoferr=0;
for i=1:N,
  % demodulator output
  if (dsource(i)==0),
    r0c=sqrt(Eb)*cos(phi)+gngauss(sgma);
    r0s=sqrt(Eb)*sin(phi)+gngauss(sgma);
    r1c=gngauss(sgma);
    r1s=gngauss(sgma);
  else
    r0c=gngauss(sgma);
    r0s=gngauss(sgma);
    r1c=sqrt(Eb)*cos(phi)+gngauss(sgma);
    r1s=sqrt(Eb)*sin(phi)+gngauss(sgma);
  end;
  % square-law detector outputs
  r0=r0c^2+r0s^2;
  r1=r1c^2+r1s^2;
  % Decision is made next.
  if (r0>r1),
    decis=0;
  else
    decis=1;
  end;
  % Let's increment the error counter, if the decision is not correct.
  if (decis~=dsource(i)),
    numoferr=numoferr+1;
  end;
end;
p=numoferr/(N);
```

7.6　多载波调制和 OFDM

到目前为止，对于传输数字信息来说考虑的一直是单一载波的调制。这一节要介绍多载波调制问题。在多载波调制中，带宽为 W 的信道划分成几个（如 K 个）子信道，其中每个子信道的带宽为 $B_c = W/K$。在这 K 个子信道中，同时并同步传输着不同的信息符号。因此，传输的数据是按频分多路复用（FDM）方式工作的。

对于每个子信道，我们赋予一个载波信号，即

$$x_k(t) = \sin 2\pi f_k t, \quad k = 0, 1, \cdots, K-1 \tag{7.6.1}$$

其中，f_k 是位于第 k 个子信道的中间频率。在每个子信道上，将符号率 $1/T$ 选成等于相邻子载波的频率间隔 B_c，因此在等号持续期 T 内这些子载波是正交的，而与任意两个子载波之间的

相对相位关系无关,即

$$\int_0^T \sin(2\pi f_k t + \phi_k)\sin(2\pi f_j t + \phi_j)\mathrm{d}t = 0 \tag{7.6.2}$$

其中,$f_k - f_j = n/T, n = 1,2,\cdots$(任何相位 ϕ_k 和 ϕ_j)。利用这个约束条件就得到了正交频分多路复用(OFDM)。

一个 OFDM 系统可以设计成没有码间干扰的系统,而在单一载波系统中码间干扰是不可避免的。如果 T_s 是一个单个载波系统中的符号区间,那么具有 K 个子信道的 OFDM 系统的符号区间就是 $T = KT_s$。通过将 K 选取得足够大,可以使 OFDM 系统中的符号区间 T 远大于信道的弥散时间,从而通过适当地选取 K 值将 OFDM 系统的码间干扰减小到任意小。在这种情况下,每个子信号的带宽足够小,因此看起来好像有一个固定不变的频率响应 $C(f_k), k = 0, 1, 2, \cdots, K-1$。

在 OFDM 系统中,调制器和解调器通常都是利用基于离散傅里叶变换的一组并联调谐的滤波器柜来实现的。为了说明这种实现方式,现在考虑一个 OFDM 调制器。该调制器产生 K 个独立的子载波,其中每个子载波从某个 QAM 信号星座图中选取的符号产生。把在 K 个子信道上相应于信息符号的这些复值信号点记为 $X_k, k = 0, 1, \cdots, K-1$,因此这些信息符号 $\{X_k\}$ 就代表了一个多载波 OFDM 信号 $x(t)$ 的离散傅里叶变换(DFT)的值,其中每个载波的调制是 QAM。由于 $x(t)$ 必须是某个实值信号,它的 N DFT$\{X_k\}$ 必定满足对称性质 $X_{N-k} = X_k^*$,因此从 K 个信息符号 $\{X_k\}$,可以根据下面的定义:

$$\begin{gathered} X_{N-k} = X_k^*, \quad k = 1, 2, \cdots, K-1 \\ X_0' = \mathrm{Real}(X_0) \\ X_K' = \mathrm{Im}(X_0) \end{gathered} \tag{7.6.3}$$

创建出 $N = 2K$ 个符号。可以看出,信息符号 X_0 被剖分为两部分,两者都是实值的。将这个新的符号序列记为 $\{X_k', k = 0, 1, \cdots, N-1\}$ 是很方便的,那么 $\{X_k'\}$ 的 N 点逆 DFT(IDFT)得到的实值序列为:

$$x_n = \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1} X_k' \mathrm{e}^{\mathrm{j}2\pi n \frac{k}{N}}, \quad n = 0, 1, \cdots, N-1 \tag{7.6.4}$$

其中,$\frac{1}{\sqrt{N}}$ 只是一个标量因子。序列 $\{x_n, 0 \leqslant n \leqslant N-1\}$ 相应于子多载波 OFDM 信号 $x(t)$ 的样本,它们由 $K = N/2$ 个子载波组成,这些可以表示为:

$$x(t) = \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1} X_k' \mathrm{e}^{\mathrm{j}2\pi \frac{kt}{T}} \quad 0 \leqslant t \leqslant T \tag{7.6.5}$$

其中,T 是信号的持续期(也就是信号区间),而 $x_n = x\left(\frac{nT}{N}\right), n = 0, 1, \cdots, N-1$。值得注意的是,信息信号 X_0 通过式(7.6.3)给出的 X_0' 和 X_K' 表示,相应于式(7.6.5)中的直流(dc)分量($f_0 = 0$)。为简单起见,我们置 $X_0 = 0$,使得由式(7.6.5)给出的多载波 OFDM 信号没有直流分量。这样,使用由式(7.6.3)给出的对称条件,式(7.6.5)中的这个多载波 OFDM 信号可以表示为:

$$x(t) = \frac{2}{\sqrt{N}}\sum_{k=1}^{K-1} |X_k| \cos\left(\frac{2\pi kt}{T} + \theta_k\right), \quad 0 \leqslant t \leqslant T \tag{7.6.6}$$

其中的信息符号是 $X_k = |X_k|\mathrm{e}^{\mathrm{j}\theta_k}, k = 1, 2, \cdots, K-1$。

解说题

解说题 7.9　[OFDM 信号的产生]

利用图 7.19 所示的 16 点 QAM 信号星座图，准随机地选取每个信息符号 $X_1, X_2, \cdots, X_9$。用 $T = 100$ s 产生 $t = 0, 1, \cdots, 100$ 的由式(7.6.5)给出的发送信号波形 $x(t)$，并画出这个波形。然后，根据式(7.6.4)计算 $n = 0, 1, \cdots, N-1$ 时的 IDFT 值 x_n。解释说明当 $Tn/N, n = 0, 1, \cdots, N-1$ 时求出的 $x(t)$ 值相应于 IDFT 值。最后，利用这些 IDFT 值 $\{x_n, 0 \leqslant n \leqslant N-1\}$，按下面的定义：

$$X_k = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x_n \mathrm{e}^{-\mathrm{j}2\pi k \frac{n}{N}} \quad k = 0, 1, \cdots, N-1 \tag{7.6.7}$$

计算 DFT，并据此展示如何从 $x(t)(t = nT/N, 0 \leqslant n \leqslant N-1)$ 的样本中恢复出信息符号 $\{X_k, 1 \leqslant k \leqslant 9\}$。

题　解

在这个例子中 $K = 10, N = 20$。本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 7.9.
echo on
K=10;N=2*K;T=100;
a=rand(1,36);
a=sign(a-0.5);
b=reshape(a,9,4);
% Generate the 16 QAM points.
XXX=2*b(:,1)+b(:,2)+j*(2*b(:,3)+b(:,4));
XX=XXX';
X=[0 XX 0 conj(XX(9:-1:1))];
xt=zeros(1,101);
for t=0:100
  for k=0:N-1
    xt(1,t+1)=xt(1,t+1)+1/sqrt(N)*X(k+1)*exp(j*2*pi*k*t/T);
    echo off
  end
end
echo on
xn=zeros(1,N);
for n=0:N-1
  for k=0:N-1
    xn(n+1)=xn(n+1)+1/sqrt(N)*X(k+1)*exp(j*2*pi*n*k/N);
    echo off
  end
end
echo on
pause % Press any key to see a plot of x(t).
plot([0:100],abs(xt))
% Check the difference between xn and samples of x(t).
for n=0:N-1
  d(n+1)=xt(T/N*n+1)-xn(1+n);
  echo off
end
echo on
```

```
e=norm(d);
Y=zeros(1,10);
for k=1:9
  for n=0:N-1
    Y(1,k+1)=Y(1,k+1)+1/sqrt(N)*xn(n+1)*exp(-j*2*pi*k*n/N);
    echo off
  end
end
echo on
dd=Y(1:10)-X(1:10);
ee=norm(dd);
```

正如上面所看到的,一个 OFDM 系统的调制器可以通过计算由式(7.6.4)给出的逆 DFT (IDFT)来实现;而从已接收信号样本中恢复信息符号$\{X_k\}$的解调器是通过计算由式(7.6.7)给出的 DFT 来实现的。代表在发送端和接收端实现这些运算的方框图见图 7.32。有关检测器的问题稍后讨论。当子载波的数目较大时(如 $K>30$),在 OFDM 系统中的调制器和解调器都能利用快速傅里叶变换算法(FFT 算法)计算 DFT 和 IDFT,而在 OFDM 系统内予以高效地实现。

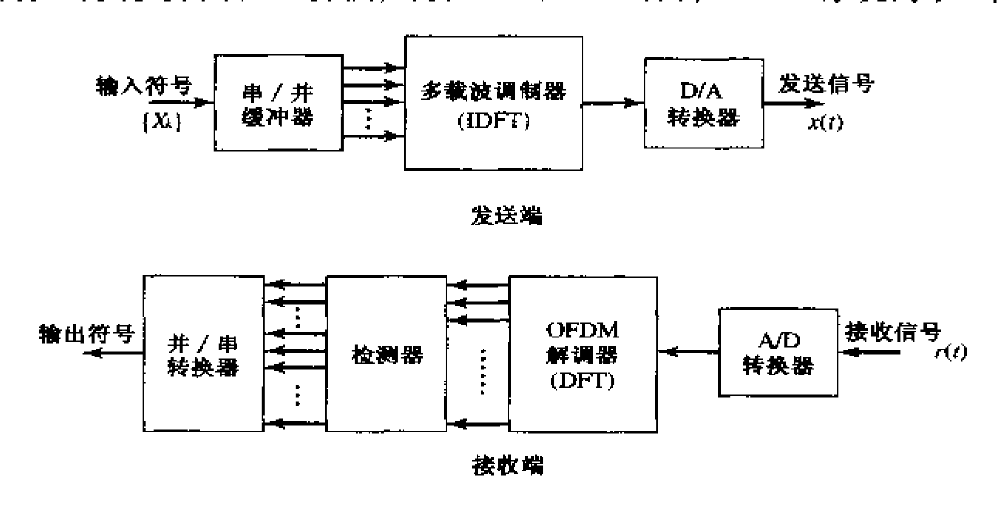


图 7.32 OFDM 发送端和接收端方框图

如果子载波的数目足够大,以至于在子信道内的时间弥散效应可以忽略,那么经由每个子信道传输的信号就只受到衰减和相移,其大小由对应的每个子信道的频率响应所确定。另外,接收到的信号受到了加性噪声的污损,因此在 OFDM 解调器输出端接收到的符号可以表示为:

$$\hat{X}_k = C_k X'_k + \eta_k \tag{7.6.8}$$

其中,$C_k = C(f_k)$是第 k 个子信道的频率响应(幅度和相位),而 η_k 是污损第 k 个符号的加性噪声。跟在 OFDM 解调器后面的检测器必须通过将每个接收符号 $\hat{X}_k$ 除以 C_k,消除信道频率响应特性的影响。一般情况下,实际做法是通过测量每个子信道的频率响应特性并利用这些测量(C_k 的估值)来消除接收符号中子信道的幅度和相位的影响。因此,在检测器上经补偿后的接收符号可以表示为:

$$\hat{X}'_k = X_k + \eta_k, \quad k = 0,1,\cdots,N-1 \tag{7.6.9}$$

一旦观察到每个接收符号,检测器就会将 $\hat{X}'_k$ 与 QAM 信号星座图中的每个可能的传输信号点进行比较,并选取星座图中在距离上最接近 $\hat{X}'_k$ 的符号。

在以上讨论中用同一个 QAM 信号星座图产生了所有子载波的信息符号。然而，OFDM 也可以在每个子载波上传输不同的比特/符号数。具体来说，由于较低衰减的子信道调制成了携带更多的比特/符号数，因此可以在不同的子载波上采用不同星座图规模的 QAM。

上面所讨论的在每个子载波上采用 QAM 的 OFDM 系统已经在各种不同应用中付诸实施了，其中包括在电话线上的高速数据传输，例如数字用户线。这种 OFDM 调制型式也称**离散多音(DMT)调制**。在欧洲和世界其他各地的数字音频广播和数字无线局域网(LAN)中也有应用 OFDM 系统的。

解说题

解说题 7.10　[加性噪声对 OFDM 的影响]

考虑在解说题 7.9 中描述的 OFDM 系统。假定由式(7.6.4)给出的传输信号的 IDFT 样本受到零均值、方差为 σ^2 的高斯噪声的污损，那么接收到的信号样本就可以表示为：

$$r_n = x_n + g_n \quad n = 0,1,\cdots,N-1$$

其中，g_n 是噪声分量。当噪声方差 $\sigma^2 = 1,2$ 和 4 时，对于每个 σ^2 值计算序列 $\{r_n\}$ 的 DFT，然后求得接收符号 $\{\hat{X}_k\}$ 的估值，对序列 $\hat{X}_k$ 进行检测，由此从检测器得到输出符号 $\{\tilde{X}_k\}$。试讨论检测到的符号的准确性。

题　解

本题的 MATLAB 脚本如下所示。这个脚本针对 $\sigma^2 = 1$ 计算接收符号的差错数。在这个脚本中改变变量“variance”(方差)就能得到不同噪声方差值的差错数。

m 文件

```
% MATLAB script for Illustrative Problem 7.10.
echo on
K=10;N=2*K;T=100;variance=1;
noise=sqrt(variance)*randn(1,N);
a=rand(1,36);
a=sign(a-0.5);
b=reshape(a,9,4);
% Generate the 16 QAM points.
XXX=2*b(:,1)+b(:,2)+j*(2*b(:,3)+b(:,4));
XX=XXX';
X=[0 XX 0 conj(XX(9:-1:1))];
x=zeros(1,N);
for n=0:N-1
  for k=0:N-1
    x(n+1)=x(n+1)+1/sqrt(N)*X(k+1)*exp(j*2*pi*n*k/N);
    echo off
  end
end
echo on
r=x+noise;
Y=zeros(1,10);
for k=1:9
  for n=0:N-1
    Y(1,k+1)=Y(1,k+1)+1/sqrt(N)*r(n+1)*exp(-j*2*pi*k*n/N);
    echo off
  end
```

```
end
echo on
% Detect the nearest neighbor in the 16 QAM constellation.
for k=1:9
  if real(Y(1,k+1))>0
    if real(Y(1,k+1))>2
      Z(1,k+1)=3;
    else
      Z(1,k+1)=1;
    end
  else
    if real(Y(1,k+1))<-2
      Z(1,k+1)=-3;
    else
      Z(1,k+1)=-1;
    end
  end
  if imag(Y(1,k+1))>0
    if imag(Y(1,k+1))>2
      Z(1,k+1)=Z(1,k+1)+3*j;
    else
      Z(1,k+1)=Z(1,k+1)+j;
    end
  else
    if imag(Y(1,k+1))<-2
      Z(1,k+1)=Z(1,k+1)-3*j;
    else
      Z(1,k+1)=Z(1,k+1)-j;
    end
  end
  echo off
end
echo on
error=max(size(find(Z(1,2:10)-X(1,2:10))));
```

7.7 通信系统中的同步

在3.3节中曾经讨论过AM信号的解调过程,特别指出了可以将解调方法分为**相干**和**非相干**方法。在**相干**解调中,将AM信号乘以与载波具有同频率、同相位的正弦信号,然后解调。在非相干解调中(仅适用于常规AM调制信号)采用包络解调,在接收端不需要对载波频率和相位进行精确的跟踪。另外,在解说题3.6中也曾经指出,在相干解调中正确的相位同步是最重要的,而相位误差会导致很大的性能变坏。

在这一章中我们讨论了数字载波调制系统的各种解调方法。在PAM,PSK和QAM解调中都曾经假设对载波频率和相位有了完全的了解。

这一节要讨论在解调器端产生与载波具有同频率、同相位的正弦信号的各种方法。这些方法是在**载波同步**这样一个论题下研究的,它适用于模拟和数字载波调制系统(在第3章和本章中已分别讨论过)。另一种同步类型称为**定时同步**、**时钟同步**或**定时恢复**,它仅在数字通信系统中会遇到。这一节也将对这类同步问题进行简要讨论。

7.7.1 载波同步

一个载波同步系统由一个本地振荡器组成，要控制该振荡器的相位，使它与载波信号同步。用一个**锁相环**(PLL)可实现这一点。锁相环是一个非线性反馈控制系统，用于控制本地振荡器的相位。在下面的讨论中，为简单起见，我们仅考虑二进制 PSK 调制系统。

PLL 被一个在载波频率(或它的倍数)的正弦信号驱动。为了得到这个驱动 PLL 的正弦信号，现将 DSB 已调信号

$$u(t) = A_c m(t)\cos(2\pi f_c t - \phi(t)) \tag{7.7.1}$$

求平方，其中 $m(t) = \pm 1$，可得：

$$\begin{aligned} u^2(t) &= A_c^2 m^2(t)\cos^2(2\pi f_c t - \phi(t)) \\ &= \frac{A_c^2}{2} m^2(t) + \frac{A_c^2}{2} m^2(t)\cos(4\pi f_c t - 2\phi(t)) \\ &= \frac{A_c^2}{2} + \frac{A_c^2}{2}\cos(4\pi f_c t - 2\phi(t)) \end{aligned} \tag{7.7.2}$$

显然，信号 $u^2(t)$有 $2f_c$ 的分量。我们之所以不直接处理 $u(t)$，是因为通常这个过程 $m(t)$是零均值的，所以 $u(t)$在 f_c 的功率含量是零。现在，如果信号 $u^2(t)$通过一个调谐在 $2f_c$ 的带通滤波器，那么输出就是中心频率为 $2f_c$，相位为 $-2\phi(t)$和幅度为 $A_c^2 H(2f_c)/2$ 的正弦信号。不失一般性，可以假设幅度为 1；也就是说，PLL 的输入是：

$$r(t) = \cos(4\pi f_c t - 2\phi(t)) \tag{7.7.3}$$

PLL 由一个乘法器、一个环路滤波器和一个电压控制振荡器(VCO)组成，如图 7.33 所示。如果假定 VCO 的输出是 $\sin(4\pi f_c t - 2\hat{\phi}(t))$，那么在环路滤波器的输入则有：

$$\begin{aligned} e(t) &= \cos(4\pi f_c t - 2\phi(t))\sin(4\pi f_c t - 2\hat{\phi}(t)) \\ &= \frac{1}{2}\sin(2\hat{\phi}(t) - 2\phi(t)) + \frac{1}{2}\sin(8\pi f_c t - 2\phi(t) - 2\hat{\phi}(t)) \end{aligned} \tag{7.7.4}$$

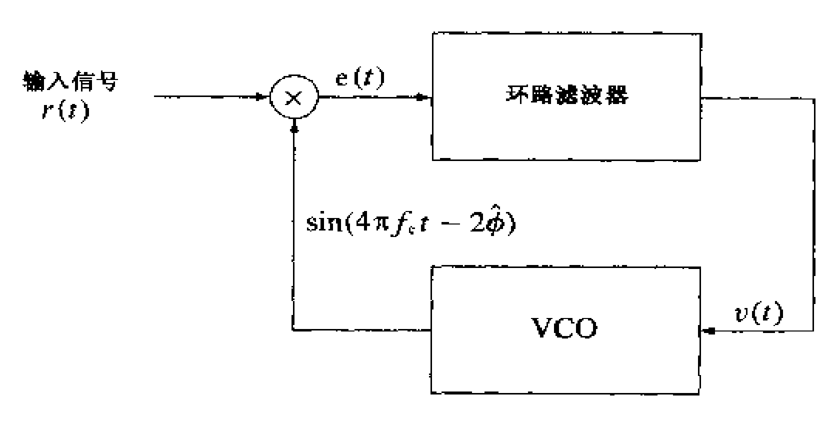


图 7.33　锁相环

注意，$e(t)$中有一个高频分量和一个低频分量。这个环路滤波器的作用就是滤除这个高频分量，并确保 $\hat{\phi}(t)$紧紧跟随 $\phi(t)$的变化。一个简单的环路滤波器是一个一阶低通滤波器，其传递函数为：

$$G(s)=\frac{1+\tau_1 s}{1+\tau_2 s} \tag{7.7.5}$$

其中，$\tau_2 >> \tau_1$。若将 VCO 的输入记为 $v(t)$，那么 VCO 的输出是一个正弦信号，它的瞬时频率偏离 $2f_c$ 的量正比于 $v(t)$。但是，VCO 输出的瞬时频率是：

$$2f_c+\frac{1}{\pi}\frac{\mathrm{d}}{\mathrm{d}t}\hat{\phi}(t)$$

因此，

$$\frac{\mathrm{d}}{\mathrm{d}t}\hat{\phi}(t)=\frac{K}{2}v(t) \tag{7.7.6}$$

或者等效为：

$$2\hat{\phi}(t)=K\int_{-\infty}^{t}v(t)\mathrm{d}\tau \tag{7.7.7}$$

其中，K 是某个比例常数。在滤除 2 次和 4 次谐波以后，这个 PLL 环路就变成了图 7.34 所示的锁相环。

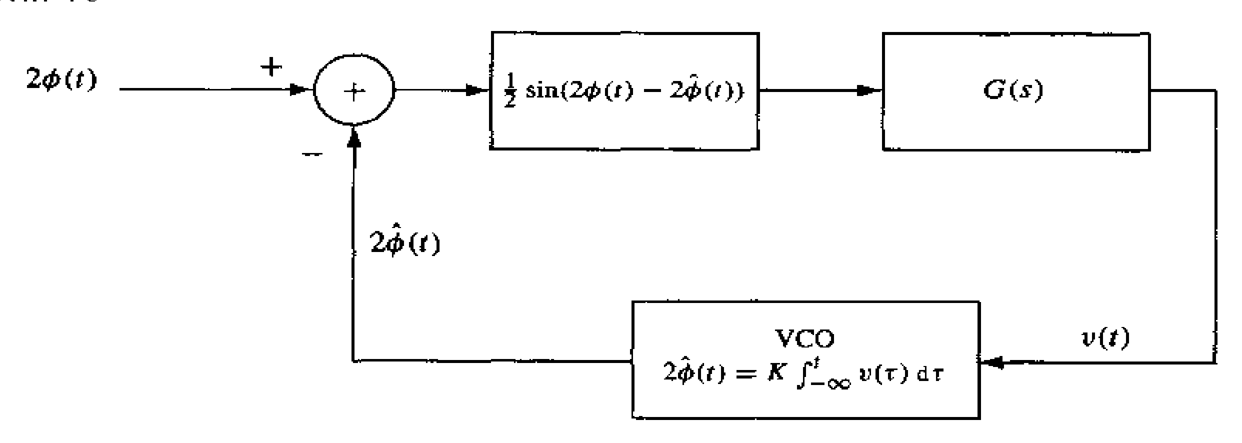


图 7.34　滤除高频分量后的锁相环

假定 $\hat{\phi}(t)$已紧随 $\phi(t)$的变化，那么 $2\phi(t)-2\hat{\phi}(t)$就很小，这样就可以使用近似式：

$$\frac{1}{2}\sin(2\phi(t)-2\hat{\phi}(t))\approx\phi(t)-\hat{\phi}(t) \tag{7.7.8}$$

利用这个近似关系之后，图 7.34 中的惟一非线性环节就可以用一个线性环节来代替，这样就得到了一个**线性化的 PLL 模型**，如图 7.35 所示。应该注意，这个模型是用变换域表示的，其中的积分器用它的等效变换域关系 $1/s$ 代替。

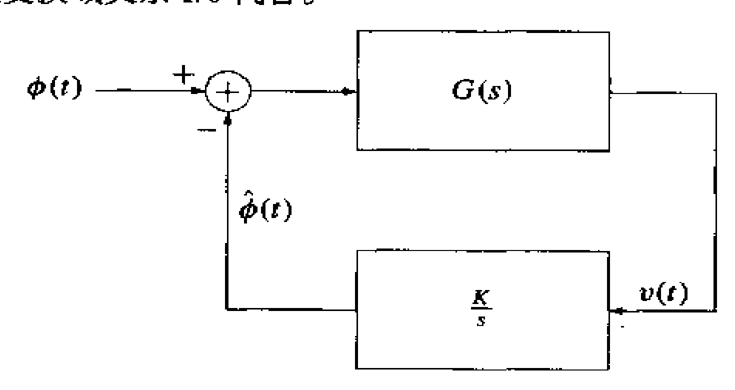


图 7.35　锁相环的线性化模型

图 7.35 中的这个模型是一个线性控制系统，其正向增益为 $G(s)$，反馈增益为 K/s，因此该系统的传递函数为：

$$H(s)=\frac{\hat{\Phi}(s)}{\Phi(s)}=\frac{KG(s)/s}{1+KG(s)/s} \tag{7.7.9}$$

按前面的假设，$G(s)$用一阶模型，

$$G(s)=\frac{1+\tau_1 s}{1+\tau_2 s} \tag{7.7.10}$$

$H(s)$则为：

$$H(s)=\frac{1+\tau_1 s}{1+(\tau_1+1/K)s+\tau_2 s^2/K} \tag{7.7.11}$$

有了传递函数 $H(s)$以后，如果 PLL 环路的输入是 $\Phi(s)$，误差就是：

$$\begin{aligned}\Delta\Phi(s)&=\Phi(s)-\hat{\Phi}(s)\\&=\Phi(s)-\Phi(s)H(s)\\&=[1-H(s)]\Phi(s)\\&=\frac{(1+\tau_2 s)s}{K+(1+K\tau_1)s+\tau_2 s^2}\Phi(s)\end{aligned} \tag{7.7.12}$$

现在假定在到达某一时刻有 $\phi(t)\approx\hat{\phi}(t)$，所以 $\Delta\phi(t)\approx 0$。在这个时刻，有某个突然变化引起了在 $\phi(t)$上的跳变，可以将它仿真成某个阶跃函数，即 $\Phi(s)=K_1/s$。由于这个变化，则有：

$$\begin{aligned}\Delta\Phi(s)&=\frac{(1+\tau_2 s)s}{K+(1+K\tau_1)s+\tau_2 s^2}\frac{K_1}{s}\\&=\frac{K_1(1+\tau_2 s)}{K+(1+K\tau_1)s+\tau_2 s^2}\end{aligned} \tag{7.7.13}$$

现在，利用拉普拉斯变换的终值定理，即

$$\lim_{t\to\infty}f(t)=\lim_{s\to 0}sF(s) \tag{7.7.14}$$

只要 $sF(s)$的全部极点都有负的实部，就可以得出：

$$\begin{aligned}\lim_{t\to\infty}\Delta\phi(t)&=\lim_{s\to 0}s\Phi(s)\\&=\lim_{s\to 0}\frac{K_1 s(1+\tau_2 s)}{K+(1+K\tau_1)s+\tau_2 s^2}\\&=0\end{aligned} \tag{7.7.15}$$

换句话说，一个简单的一阶环路滤波器会导致 PLL 能跟踪在输入相位上的阶跃变化。

式(7.7.11)的传递函数可以写成如下标准形式：

$$H(s)=\frac{(2\zeta\omega_n-\omega_n^2/K)s+\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2} \tag{7.7.16}$$

其中，

$$\omega_n=\sqrt{\frac{K}{\tau_2}}$$

$$\zeta=\frac{\omega_n(\tau_1+1/K)}{2}$$

ω_n 是自然频率，ζ 是阻尼系数。

解说题

解说题 7.11 [一阶 PLL]

假设

$$G(s)=\frac{1+0.01\ s}{1+s}$$

和 $K=1$,求出并画出该 PLL 对输入相位单位阶跃变化的响应。

题 解

由于 $\tau_1=0.01,\tau_2=1$,因此

$$\omega_n=1$$
$$\zeta=0.505$$

这样就得到:

$$H(s)=\frac{0.01\ s+1}{s^2+1.01\ s+1}$$

于是对 $\phi(t)=u(t)$(即 $\Phi(s)=1/s$)的响应为:

$$\hat{\Phi}(s)=\frac{0.01s+1}{s^3+1.01s^2+s+1}$$

为了求出并画出 $\hat{\phi}(t)$对输入 $u(t)$的时间响应,我们注意到,必须求出传递函数为 $H(s)$ 的系统对输入 $u(t)$的输出。最容易的方法是使用状态空间方法。使用 MATLAB 函数 tf2ss.m,即可获得由传递函数描述的系统的状态空间模型。求出该系统的状态空间表示之后,就得到了数值解的阶跃响应。

函数 tf2ss.m 取传递函数 $H(s)$的分子和分母,得到它的状态空间表示 $\boldsymbol{A},\boldsymbol{B},\boldsymbol{C}$ 和$\boldsymbol{D}$,其形式为:

$$\begin{cases}\dfrac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{x}(t)=\boldsymbol{A}\boldsymbol{x}(t)+\boldsymbol{B}u(t)\\ y(t)=\boldsymbol{C}\boldsymbol{x}(t)+\boldsymbol{D}u(t)\end{cases}$$

这一表示方式可近似为:

$$\begin{cases}\boldsymbol{x}(t+\Delta t)=\boldsymbol{x}(t)+\boldsymbol{A}\boldsymbol{x}(t)\Delta t+\boldsymbol{B}u(t)\Delta t\\ y(t)=\boldsymbol{C}\boldsymbol{x}(t)+\mathrm{D}u(t)\end{cases}$$

或等效为:

$$\begin{cases}\boldsymbol{x}(i+1)=\boldsymbol{x}(i)+\boldsymbol{A}\boldsymbol{x}(i)\Delta t+\boldsymbol{B}u(i)\Delta t\\ y(i)=\boldsymbol{C}\boldsymbol{x}(i)+\boldsymbol{D}u(i)\end{cases}$$

对于这个解说题,分别选 $u(t)$为阶跃函数,$H(s)$的分子和分母向量为[0.01 1]和[1 1.01 1]就足够了。使用选取的这个分子和分母向量,该系统的状态空间参数是:

$$\boldsymbol{A}=\begin{bmatrix}-1.01 & -1\\ 1 & 0\end{bmatrix}$$

$$\boldsymbol{B}=\begin{bmatrix}1\\ 0\end{bmatrix}$$

$$\boldsymbol{C}=[0.01\quad 1]$$

$$\boldsymbol{D}=0$$


这个 PLL 环路的输出如图 7.36 所示。

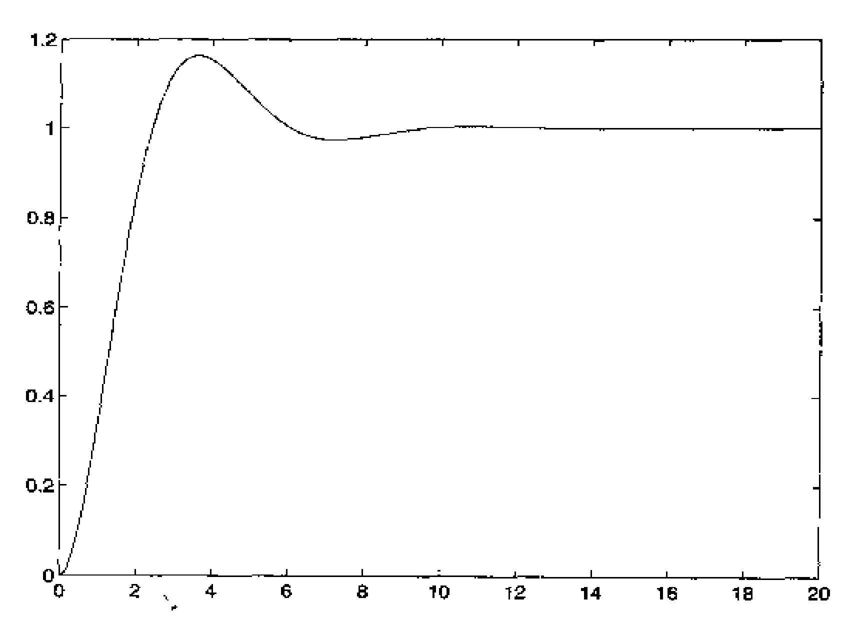


图 7.36　在解说题 7.11 中，PLL 对输入相位突然变化的响应

由图 7.36 可见，这个 PLL 的输出最终还是跟上了输入。然而，跟踪输入的速度取决于环路滤波器的参数和压控振荡器 VCO 的比例常数 K。

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 7.11
echo on
num=[0.01 1];
den=[1 1.01 1];
[a,b,c,d]=tf2ss(num,den);
dt=0.01;
u=ones(1,2000);
x=zeros(2,2001);
for i=1:2000
  x(:,i+1)=x(:,i)+dt.*a*x(:,i)+dt.*b*u(i);
  y(i)=c*x(:,i);
  echo off;
end
echo on;
t=[0:dt:20];
plot(t(1:2000),y)
```

7.7.2　时钟同步

在第 5 章和这一章中，我们已经知道，最佳接收机的一种常见实现是使用匹配滤波器和在匹配滤波器输出端的采样器。在所有这些情况下，都假定接收机对采样瞬间具有完全的了解，并能在这个时刻精确采样。在发射机和接收机之间完成这类同步的系统称为**定时恢复**、**时钟同步**或**符号同步**系统。

一种简单的时钟同步实现是采用早-迟门(early-late gate)。早-迟门的工作原理基于PAM通信系统中的这样一个事实;匹配滤波器的输出是PAM系统中所用基本脉冲信号的自相关函数(可能有某个延时),这个自相关函数在最佳采样时刻是最大的,而且是对称的。这就意味着当无噪声存在时,在采样时刻 $T^+ = T + \delta$ 和 $T^- = T - \delta$,采样器的输出是相等的,即

$$y(T^+) = y(T^-) \tag{7.7.17}$$

很明显,这种情况下的最佳采样时间就是提前和迟后采样时间之间的中点:

$$T = \frac{T^+ + T^-}{2} \tag{7.7.18}$$

现在假设不是在最佳采样时间 T 采样,而是在 T_1 采样。如果在 $T^+ = T_1 + \delta$ 和 $T^- = T_1 - \delta$,取两个额外的样本,这两个样本对于最佳采样时间 T 就不是对称的了,因此不再相等。对于正和负输入脉冲的典型自相关函数及其3个样本,如图7.37所示。
图中的

$$T^- = T - \delta_1$$
$$T^+ = T + \delta_2$$

其中,

$$\delta_1 < \delta_2 \tag{7.7.19}$$

如图所示即可得出:

$$|y(T^-)| > |y(T^+)| \tag{7.7.20}$$

同时,在这种情况下有:

$$T < T_1 = \frac{T^- + T^+}{2} \tag{7.7.21}$$

因此,当 $|y(T^-)| > |y(T^+)|$ 时,正确的采样时间是在假定的采样时间之前,采样应该提前一点进行。相反,当 $|y(T^-)| < |y(T^+)|$ 时,采样时间应该迟后一点。显然,当 $|y(T^-)| = |y(T^+)|$ 时,采样时间是正确的,不需要进行校正。

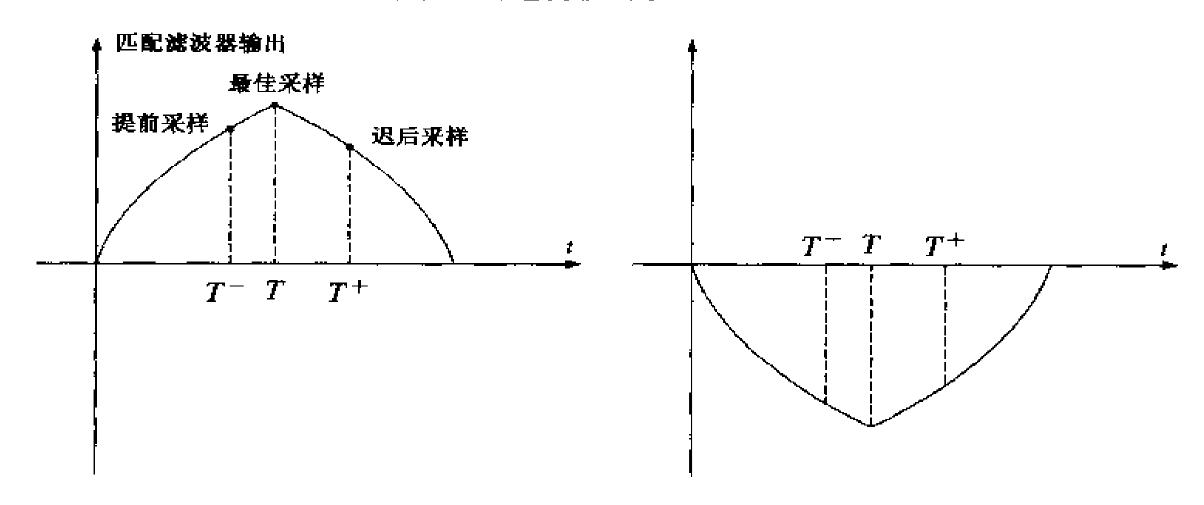


图7.37 匹配滤波器输出和提前与迟后采样

因此,早-迟门同步系统取 T_1, $T^- = T_1 - \delta$ 和 $T^+ = T_1 + \delta$ 这3个样本,然后比较 $|y(T^-)|$ 和 $|y(T^+)|$,根据它们的相对值,产生一个校正采样时间的信号。

解说题

解说题 7.12　[时钟同步]

一个二进制 PAM 通信系统使用滚降系数 $\alpha=0.4$ 的升余弦波形，系统传输速率是 4800 bps。写一个 MATLAB 文件，仿真这个系统的早－迟门的工作。

题　解

因为速率是 4800 bps，所以有：

$$T=\frac{1}{4800} \tag{7.7.22}$$

$\alpha=0.4$，升余弦波形的表达式变为：

$$\begin{aligned} x(t) &= \mathrm{sinc}(4800t)\frac{\cos(4800\times 0.4\pi t)}{1-4\times 0.16\times 4800^2 t^2} \\ &= \mathrm{sinc}(4800t)\frac{\cos(1920\pi t)}{1-1.4746\times 10^7 t^2} \end{aligned} \tag{7.7.23}$$

很明显，这个信号从 $-\infty$ 延伸至 $+\infty$。图 7.38 是该信号的图。

从图 7.38 可以很清楚地看到，实际上仅考虑区间 $|t|\leqslant 0.6\times 10^{-3}$ s 就足够了，这大致上相当于 $[-3T,3T]$。以这个区间截断这个升余弦并计算自相关函数，即可得到图 7.39 所示的自相关函数的波形。

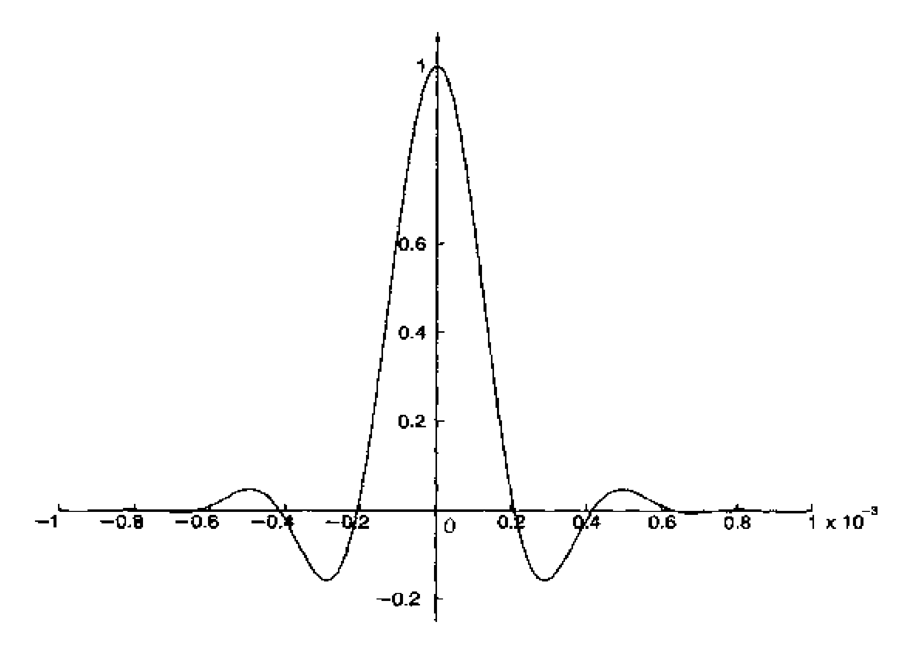


图 7.38　解说题 7.12 中的升余弦信号

在下面给出的 MATLAB 脚本中，首先计算出升余弦信号和自相关函数，并画出它们的波形。在这个特例中，自相关函数的长是 1201，最大点（即最佳采样时间）发生在第 600 个分量上。我们对两种情况进行测试，一种是当不正确采样时间在 700 时，另一种是在 500 时。在这两种情况下，早－迟门都将采样时间校正到最佳时间 600。

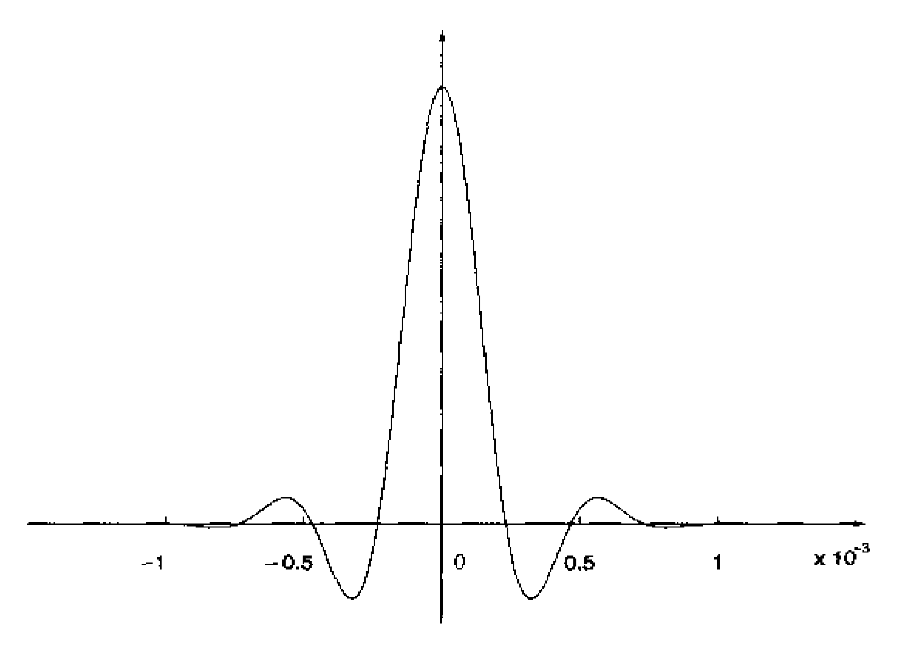


图 7.39 升余弦信号的自相关函数

m 文件

```
% MATLAB script for Illustrative Problem 7.12
echo on
alpha=0.4;
T=1/4800;
t=[-3*T:1.001*T/100:3*T];
x=sinc(t./T).*(cos(pi*alpha*t./T)./(1-4*alpha^2*t.^2/T^2));
pause % Press any key to see a plot of x(t).
plot(t,x)
y=xcorr(x);
ty=[t-3*T,t(2:length(t))+3*T];
pause % Press any key to see a plot of the autocorrelation of x(t).
plot(ty,y);
d=60;                                   % Early and late advance and delay
ee=0.01;                                % Precision
e=1;                                    % Step size
n=700;                                  % The incorrect sampling time
while abs(abs(y(n+d))-abs(y(n-d)))>=ee
  if abs(y(n+d))-abs(y(n-d))>0
    n=n+e;
  elseif abs(y(n+d))-abs(y(n-d))<0
    n=n-e;
  end
  echo off ;
end
echo on ;
pause % Press any key to see the corrected sampling time
n
n=500;                                  % Another incorrect sampling time
while abs(abs(y(n+d))-abs(y(n-d)))>=ee
  if abs(y(n+d))-abs(y(n-d))>0
    n=n+e;
  elseif abs(y(n+d))-abs(y(n-d))<0
    n=n-e;
```

```
  end
  echo off ;
end
echo on ;
pause % Press any key to see the corrected sampling time
n
```

7.8　习题

7.1　在载波幅度调制 PAM 系统中，发送滤波器有一个滚降系数 $\alpha = 1$ 的开方升余弦谱特性，载波频率 $f_c = 40/T$。求出并画出该基带信号和幅度调制 PAM 信号的频谱。

7.2　当载波频率 $f_c = 80/T$ 时，重做习题 7.1。

7.3　当发送滤波器有开方双二进制频谱特性时，重做习题 7.1。

7.4　本题的目的是要通过用 MATLAB 的数值解计算式(7.2.9)，以说明式(7.2.9)和式(7.2.10)都成立。假设脉冲 $g_T(t)$是矩形，即

$$g_T(t) = \begin{cases} 1, & 0 \leqslant t \leqslant 2 \\ 0, & \text{其余 } t \end{cases}$$

令载波频率 $f_c = 2000$ Hz。在由式(7.2.6)给出的信号波形 $\psi(t)$上用采样率 $F_s = 20\,000$样本/s 进行采样，并用下面的和式：

$$\frac{1}{N}\sum_{n=0}^{N-1}\psi^2(nT_s) = \frac{1}{N}\sum_{n=0}^{N-1}\psi^2\left(\frac{n}{F_s}\right)$$

近似式(7.2.8)的积分计算 $\psi(t)$的能量，其中 $N = 40\,000$ 个样本。写一个 MATLAB 程序，产生样本 $\psi(n/F_s)$，并按上述过程完成该信号能量的计算。

7.5　可以用 MATLAB 数值解完成由式(7.2.14)给出的接收信号 $r(t)$和 $\psi(t)$的互相关。写一个 MATLAB 程序，计算相关器输出：

$$y(n) = \sum_{k=0}^{n} r\left(\frac{k}{F_s}\right)\psi\left(\frac{k}{F_s}\right), \quad n = 0,1,\cdots,N-1$$

其中，F_s 是采样频率。当 $r(t) = \psi(t)$时，求出并画出 $y(n)$。这里的 $\psi(t)$是习题 7.4 中所述的波形，$F_s = 20\,000$ Hz。

7.6　当信号 $g_T(t)$为

$$g_T(t) = \begin{cases} \frac{1}{2}(1-\cos(\pi t)), & 0 \leqslant t \leqslant 2 \\ 0, & \text{其余 } t \end{cases}$$

并且其他参数与习题 7.4 中的相同时，求出并画出习题 7.5 中的相关$|y(n)|$。

7.7　在解说题 7.2 中，8 相 PSK 波形有恒定的幅度。现在假设 $g_T(t)$不是矩形，而是

$$g_T(t) = \begin{cases} \frac{1}{2}(1-\cos(2\pi t/T)), & 0 \leqslant t \leqslant T \\ 0, & \text{其余 } t \end{cases}$$

写一个 MATLAB 程序，计算并画出在这种情况下当 $f_c = 6/T$ 时 $M = 8$ 的 PSK 信号波形。

7.8　写一个MATLAB程序，用数值法计算PSK信号的接收信号 $r(t)$与式(7.3.9)给出的两个

基函数的互相关。也就是计算

$$y_c(n)=\sum_{k=0}^{n} r\left(\frac{k}{F_s}\right)\psi_1\left(\frac{k}{F_s}\right),\quad n=0,1,\cdots,N-1$$

$$y_s(n)=\sum_{k=0}^{n} r\left(\frac{k}{F_s}\right)\psi_2\left(\frac{k}{F_s}\right),\quad n=0,1,\cdots,N-1$$

其中,N 是 $r(t)$,$\psi_1(t)$和 $\psi_2(t)$的样本数。当 $r(t)$和 $g_T(t)$为

$$r(t)=s_{mc}\psi_1(t)+s_{ms}\psi_2(t)$$

$$g_T(t)=\begin{cases}2, & 0\leqslant t\leqslant 2\\ 0, & 其余\ t\end{cases}$$

$f_c=1000$ Hz,$F_s=10\,000$ 样本/s,传输的信号点是:

a. $\boldsymbol{s}_m=(s_{mc},s_{ms})=(1,0)$

b. $\boldsymbol{s}_m=(-1,0)$

c. $\boldsymbol{s}_m=(0,1)$

时,求出并画出这些相关序列。

7.9 写一个 MATLAB 程序,按解说题 7.3 所述完成 $M=4$ 的 PSK 通信系统的 Monte Carlo 仿真,但是把检测器改为计算按式(7.3.16)给出的接收信号相位 θ_r,并将最接近于 θ_r 的相位的信号选为信号点。

7.10 写一个 MATLAB 程序,对 $M=4$ 的 DPSK 系统实现一个差分编码器和一个差分解码器。传送一个 2 比特符号序列通过这个编码器和解码器的级联,以验证编码器和解码器的工作情况,并证明输出序列与输入序列是一样的。

7.11 写一个 MATLAB 程序,实现一个二进制 DPSK 通信系统的 Monte Carlo 仿真。在这种情况下,传输信号相位是 $\theta=0°$和 $\theta=180°$。$\theta=0°$的相位变化对应于传输一个 0,而 $\theta=180°$的相位变化对应于传输一个 1。在不同的 SNR 参数$\mathcal{E}_b/N_0$ 值下,完成传输 $N=10\,000$ 个比特的仿真。为了方便起见,可以将$\mathcal{E}_b$ 归一化到 1。然后,有 $\sigma^2=N_0/2$,SNR 则为$\mathcal{E}_b/N_0=1/(2\sigma^2)$,其中 σ^2 是加性噪声分量的方差。所以,SNR 可以通过给加性噪声分量的方差加权予以控制。画出测出的二进制 DPSK 的误码率,并将它和式(7.3.23)给出的理论差错概率进行比较。

7.12 写一个 MATLAB 程序,对图 P7.12 给出的信号星座图产生并画出式(7.4.2)给出的 $M=8$ 的 QAM 信号波形。假设脉冲波形 $g_T(t)$是矩形,即

$$g_T(t)=\begin{cases}1, & 0\leqslant t\leqslant T\\ 0, & 其余\ t\end{cases}$$

载波频率 $f_c=8/T$。

7.13 当脉冲波形 $g_T(t)$为

$$g_T(t)=\begin{cases}\frac{1}{2}(1-\cos(2\pi t/T)), & 0\leqslant t\leqslant T\\ 0, & 其余\end{cases}$$

时,重做习题 7.12。

7.14 写一个 MATLAB 程序,对应于图 P7.12 所示的信号星座图完成一个 $M=8$ 的 QAM 通信

系统的 Monte Carlo 仿真。在不同的 SNR 参数$\mathcal{E}_{\text{avb}}/N_0$值下，对 $N=10\,000$(3 比特)个符号进行仿真。为了方便起见，将$\mathcal{E}_{\text{avb}}$归一化到 1。然后，有 $\sigma^2=N_0/2$，SNR 则为$\mathcal{E}_{\text{avb}}/N_0=1/(2\sigma^2)$，其中 σ^2 是两个加性噪声分量中每个的方差。画出测出的该 QAM 系统的符号误码率，并将它和式(7.4.12)给出的理论差错概率(上界)进行比较。

7.15　对图 P7.15 所示的 $M=8$ 的信号星座图重做习题 7.14 的仿真。试比较这两个 $M=8$ 的 QAM 信号星座图的差错概率，并指出哪个星座图给出的性能更好。

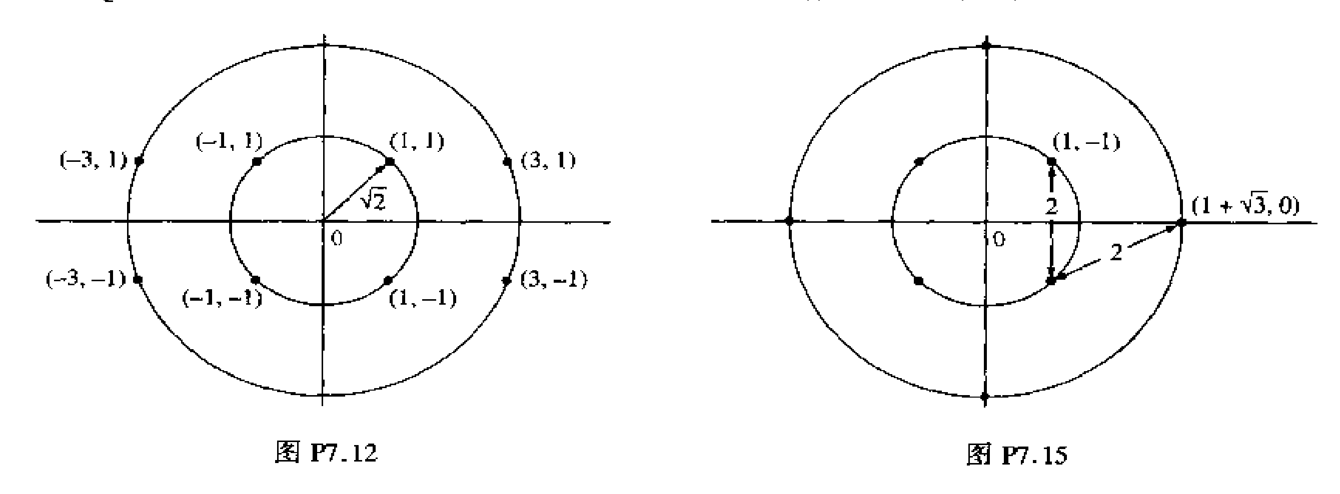


图 P7.12　　　　图 P7.15

7.16　现考虑二进制 FSK 信号：

$$u_1(t)=\sqrt{\frac{2\mathcal{E}_{\text{b}}}{T_{\text{b}}}}\cos(2\pi f_1 t),\quad 0\leqslant t\leqslant T_{\text{b}}$$

$$u_2(t)=\sqrt{\frac{2\mathcal{E}_{\text{b}}}{T_{\text{b}}}}\cos(2\pi f_2 t),\quad 0\leqslant t\leqslant T_{\text{b}}$$

$$f_2=f_1+\frac{1}{2T_{\text{b}}}$$

设 $f_1=1000/T_{\text{b}}$。通过用 $F_s=5000/T_{\text{b}}$ 采样率在比特区间 $0\leqslant t\leqslant T_{\text{b}}$ 内对这两个波形采样，得到 5000 个样本。写一个 MATLAB 程序，产生 $u_1(t)$和 $u_2(t)$的各 5000 个样本，并计算互相关：

$$\sum_{n=0}^{N-1}u_1\left(\frac{n}{F_s}\right)u_2\left(\frac{n}{F_s}\right)$$

据此用数值方法确认 $u_1(t)$和 $u_2(t)$的正交性条件。

7.17　当接收信号 $r(t)$为

$$r(t)=\cos\left(2\pi f_1 t+\frac{\pi}{2}\right),\quad 0\leqslant t\leqslant T_{\text{b}}$$

时，利用在解说题 7.7 中给出的 MATLAB 程序计算并画出相关器的输出。

7.18　当传输信号是 $u_2(t)$，接收信号 $r(t)$为：

$$r(t)=\cos\left(2\pi f_2 t+\frac{\pi}{4}\right),\quad 0\leqslant t\leqslant T$$

时，利用在解说题 7.7 中给出的 MATLAB 程序计算并画出相关器的输出。

7.19　写一个 MATLAB 程序，完成 $M=4$ 的 FSK 通信系统的仿真。采用的频率为：

$$f_k = f_1 + \frac{k}{T}, \qquad k = 0,1,2,3$$

采用的检测器为平方律检测器。在不同的 SNR 参数$\mathcal{E}_b/N_0$ 值下对 $N = 10\,000$(2 比特)个符号进行仿真,并记录符号差错和比特差错数。画出测出的符号和比特误码率,并将其与式(7.5.19)以及式(7.5.21)给出的理论符号和比特差错概率进行比较。

7.20 利用产生传输 4 个信号符号(X_1, X_2, X_3, X_4)的 OFDM 信号重做解说题 7.9。

7.21 利用产生传输 9 个信号符号$(X_1, X_2, \cdots, X_9)$的 OFDM 信号重做解说题 7.9。用 FFT 算法计算 DFT 和 IDFT。

7.22 对于传输 4 个信息符号(X_1, X_2, X_3, X_4)的 OFDM 系统,当噪声方差为 $\sigma^2 = 1, 2$ 和 4 时,重做解说题 7.10。

7.23 在解说题 7.11 中曾假定输入相位有某个突变,并且仿真结果表明:一阶环路滤波器能够跟踪这样的变化。现在假设输入相位按照斜坡函数变化,即从开始就线性增长。在这个情况下仿真一阶 PLL 的性能,并判断这个环路是否能够跟踪这样的变化。

7.24 当环路滤波器为

$$G(s) = \frac{1}{s + \sqrt{2}}$$

时,重做解说题 7.11。同时求闭环传递函数 $H(s)$及其相应的环路自然频率和阻尼系数。该环路稳定吗?

7.25 当存在 AWGN, SNR 值分别为 20 dB, 10 dB, 5 dB 和 0 dB 时,用矩形脉冲重做解说题 7.12。

第8章　信道容量和编码

8.1　概述

任何通信系统的目的都是将某个信息源产生的信息从甲地传送到乙地。传输信息经由的媒质称为**通信信道**。在第4章中我们已经知道，一个信源的信息含量是用该信源的熵来度量的，而这个熵的最常用单位是比特(bit)。我们也已经知道，一个信源的适当的数学模型是一个随机过程。

这一章要讨论适当的通信信道的数学模型，还将讨论称为**信道容量**的这样一个量。这个量是对任意通信信道定义的，并且给出了这个信道可以传输的信息量的基本限制。我们要特别考虑两类信道：二进制对称信道(BSC)和加性高斯白噪声信道(AWGN)。

本章第二部分专门讨论在通信信道上进行可靠通信的编码技术。我们将讨论两种最常见的编码方法，即分组编码和卷积编码。在本章的稍后各节中要对这两种码的编码和解码技术及其性能进行详细讨论。

8.2　信道模型和信道容量

通信信道将载有信息的信号传送到目的地。在这个传输过程中，载有信息的信号经受了各种变化。有些变化从本质上说是确定性的，例如衰减、线性和非线性失真等；而另一些变化则是概然性的，例如附加噪声、多径衰落等。由于可以把确定性变化看成是随机变化的特例，所以从最一般的意义上来说，通信信道的数学模型在输入和输出信号之间是一个随机的依赖关系。

8.2.1　信道模型

在最简单的情况下，对于一个信道，可以用每个输出和其相应的输入的关联的条件概率来建模。这样的信道称为**离散无记忆信道**(discrete-memoryless channel，简称DMC)，并且完全由信道输入字符集$\mathcal{X}$，信道输出字符集$\mathcal{Y}$和**信道转移概率矩阵** $p(y \mid x)$（对全部 $x \in \mathcal{X}$，$y \in \mathcal{Y}$已知）来表征。离散无记忆信道的一种特例是**二进制对称信道**(binary symmetric channel，简称BSC)，这种信道可以看成是在一条高斯信道上，输出为硬判定时二进制传输的一个数学模型。一条二进制对称信道对应于$\mathcal{X} = \mathcal{Y} = \{0,1\}$和 $p(y=0 \mid x=1) = p(y=1 \mid x=0) = \varepsilon$ 的情况。这种信道的一个示意性模型如图8.1所示，图中的参数 ε 称为信道的**交叉概率**(crossover probability)。

8.2.2　信道容量

按照定义，信道容量就是在这条信道上能够进行可靠信息传输的最大速率。当存在一个分组长度增长的码的序列时，随着分组长度的增长，当这种码的差错概率趋于零时，可靠传输是可能的。信道容量记为 C，按定义在速率 $R < C$ 时，在这条信道上的可靠传输是可能的；当 $R > C$ 时，可靠传输是不可能的。

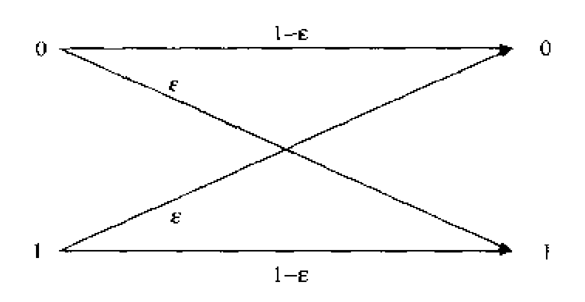


图 8.1 二进制对称信道(BSC)

香农信息论的基本结果表明,对于离散无记忆信道,它的容量由下面的表示式给出:

$$C = \max_{p(x)} I(X;Y) \tag{8.2.1}$$

其中,$I(X;Y)$代表 X(信道输入)和 Y(信道输出)之间的互信息(mutual information),其最大化是在全部信道输入概率分布上完成的。

两个随机变量 X 和 Y 之间的互信息定义为:

$$I(X;Y) = \sum_{x\in\mathscr{X}}\sum_{y\in\mathscr{Y}} p(x)p(y\mid x)\log_2\frac{p(x,y)}{p(x)p(y)} \tag{8.2.2}$$

其中,互信息以比特表示。

对于二进制对称信道,容量由下面的简单关系给出:

$$C = 1 - H_b(\varepsilon) \tag{8.2.3}$$

其中,ε 是信道的交叉概率,$H_b(\cdot)$代表二进制熵函数:

$$H_b(x) = -x\log_2(x) - (1-x)\log_2(1-x) \tag{8.2.4}$$

另一种重要的信道模型是输入功率受限的带限加性高斯白噪声信道。这种信道模型如图 8.2所示。

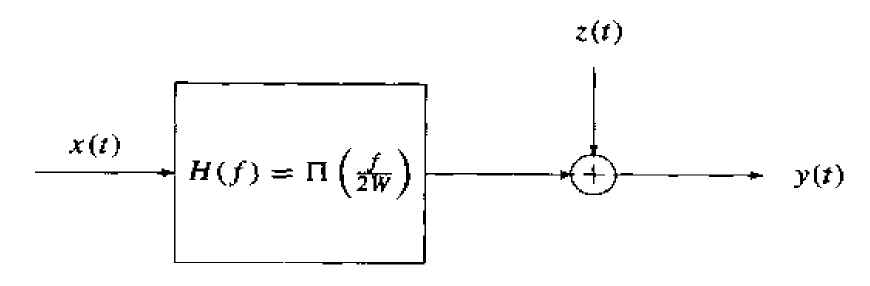


图 8.2 带限加性高斯白噪声信道

该信道带限在$[-W,W]$内,噪声是高斯型的且为白色,(双边)功率谱密度为 $N_0/2$,而信道输入是一个满足某一输入功率受限为 P 的过程。香农以 bps 表示这种信道容量,如下所示:

$$C = W\log_2\left(1+\frac{P}{N_0 W}\right) \quad \text{bps} \tag{8.2.5}$$

对于具有输入功率受限 P 和噪声方差 σ^2 的离散时间 AWGN 信道,以每次传输的比特数目表示的容量为:

$$C = \frac{1}{2}\log_2\left(1+\frac{P}{\sigma^2}\right) \tag{8.2.6}$$

解说题

解说题 8.1　[二进制对称信道容量]

利用 BPSK 信号在一个加性高斯白噪声信道上传输二进制数据,并在用最佳匹配滤波器检测的输出端利用硬判决解码。

1. 画出该信道的差错概率

$$\gamma = \frac{\mathscr{E}}{N_0} \tag{8.2.7}$$

的函数图,其中$\mathscr{E}$是在每个 BPSK 信号中的能量,而 $N_0/2$ 是噪声功率谱密度。假设 γ 在 -20 dB 到 20 dB 之间变化。

2. 画出作为 γ 的函数的所得信道容量的图

题　解

1. 使用最佳检测的 BPSK 的差错概率是:

$$p = Q(\sqrt{2\gamma}) \tag{8.2.8}$$

对应的图见图 8.3。

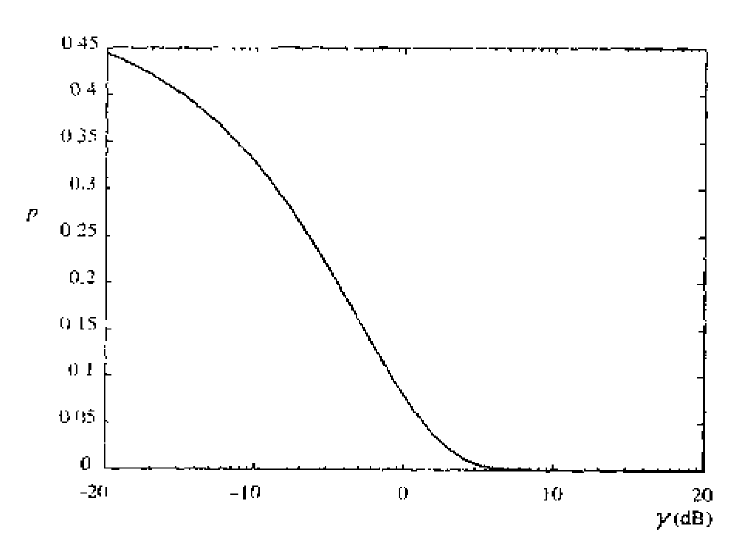


图 8.3　BPSK 差错概率与 $\gamma = \mathscr{E}/N_0$ 的函数关系

2. 现在用下面的关系:

$$\begin{aligned} C &= 1 - H_b(p) \\ &= 1 - H_b(Q(\sqrt{2\gamma})) \end{aligned} \tag{8.2.9}$$

可得 C 对于 γ 的图,如图 8.4 所示。

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 8.1.
echo on
gamma_db=[-20:0.1:20];
gamma=10.^(gamma_db./10);
p_error=q(sqrt(2.*gamma));
capacity=1.-entropy2(p_error);
pause % Press a key to see a plot of error probability vs. SNR/bit.
```

```
clf
semilogx(gamma,p_error)
xlabel('SNR/bit')
title('Error probability versus SNR/bit')
ylabel('Error Prob.')
pause % Press a key to see a plot of channel capacity vs. SNR/bit.
clf
semilogx(gamma,capacity)
xlabel('SNR/bit')
title('Channel capacity versus SNR/bit')
ylabel('Channel capacity')
```

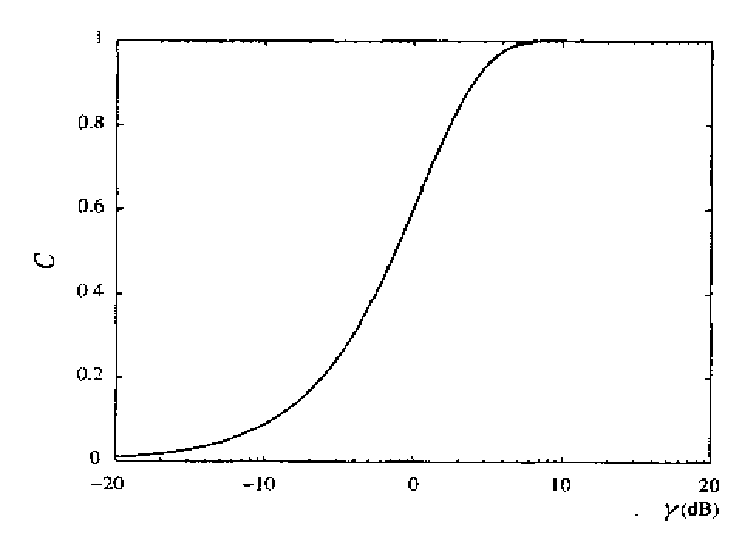


图 8.4　信道容量与 $\gamma = \mathcal{E}/N_0$ 的函数关系

解说题

解说题 8.2　[高斯信道容量]

1. 画出带宽 $W = 3000$ Hz 的加性高斯白噪声信道的容量作为 P/N_0 的函数的图，P/N_0 在 −20 dB 至 30 dB 之间变化。

2. 画出 $P/N_0 = 25$ dB 时，加性高斯白噪声信道的容量作为 W 的函数的图，尤其是当 W 无限增大时，信道容量是什么？

题　解

1. 所期望的图由图 8.5 给出。

2. 图 8.6 给出了容量作为带宽的函数。由图可见，当 P/N_0 或者 W 之一趋于零时，信道容量也趋于零。然而，当 P/N_0 或 W 趋于无限大时，容量的变化特性是不同的。当 P/N_0 趋于无限大时，如图 8.5 所示，容量也趋于无限大。然而，当 W 趋于无限大时，容量就到了某一极限值，该极限值由 P/N_0 决定。

为了确定这个极限值，有

$$\lim_{W \to \infty} W \log_2\left(1 + \frac{P}{N_0 W}\right) = \frac{P}{N_0 \ln 2} \tag{8.2.10}$$

$$= 1.4427 \frac{P}{N_0} \tag{8.2.11}$$

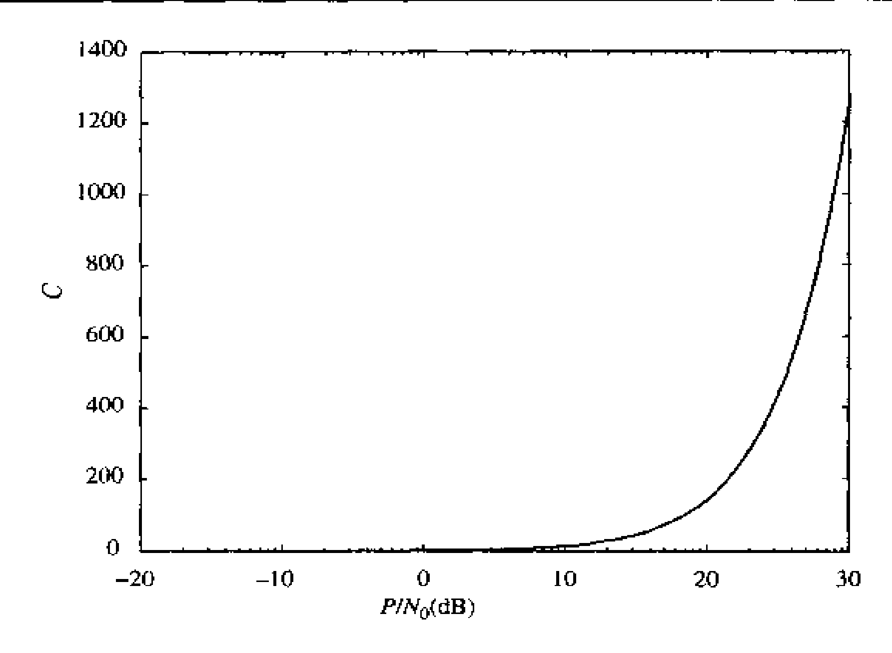


图 8.5 $W = 3000$ Hz 的 AWGN 信道容量与 P/N_0 的函数关系

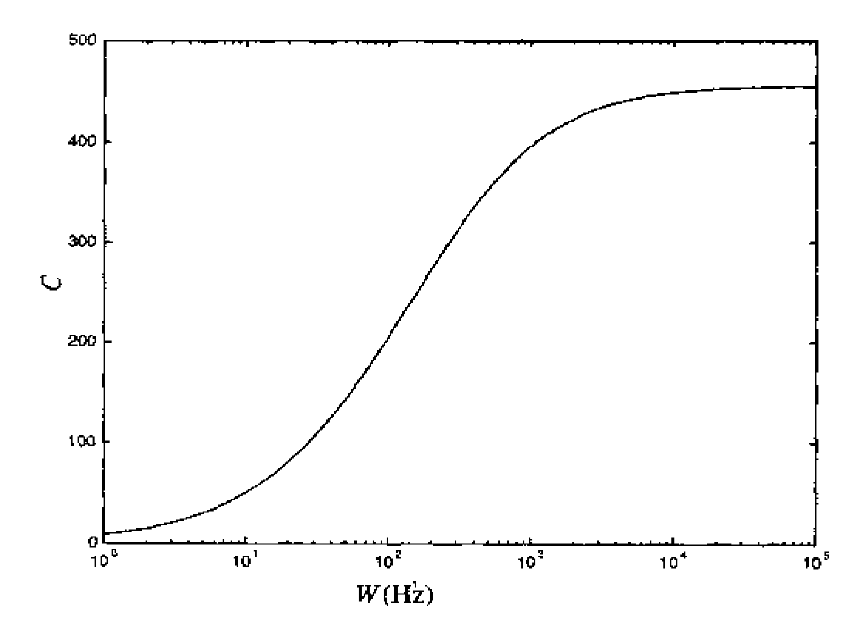


图 8.6 在 AWGN 信道中作为带宽的函数的容量

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 8.2.
echo on
pn0_db=[  20:0.1:30];
pn0=10.^(pn0_db./10);
capacity=3000.*log2(1+pn0/3000);
```

```
pause % Press a key to see a plot of channel capacity vs. P/N0.
clf
semilogx(pn0,capacity)
title('Capacity vs. P/N0 in an AWGN channel')
xlabel('P/N0')
ylabel('Capacity (bits/second)')
clear
w=[1:10,12:2:100,105:5:500,510:10:5000,5025:25:20000,20050:50:100000];
pn0_db=25;
pn0=10^(pn0_db/10);
capacity=w.*log2(1+pn0./w);
pause % Press a key to see a plot of channel capacity vs. bandwidth
clf
semilogx(w,capacity)
title('Capacity vs. bandwidth in an AWGN channel')
xlabel('Bandwidth (Hz)')
ylabel('Capacity (bits/second)')
```

解说题

解说题 8.3 [二进制输入 AWGN 信道的容量]

通过使用两个二进制输入电平 A 和 $-A$ 和方差为 σ^2 的零均值加性高斯噪声,给一个二进制输入的 AWGN 信道建模。在这种情况下,$\mathcal{X}=\{-A,A\}$,$\mathcal{Y}=\mathbb{R}$,$p(y\mid X=A)\sim\mathcal{N}(A,\sigma^2)$,$p(y\mid X=-A)\sim\mathcal{N}(-A,\sigma^2)$。画出该信道容量作为 A/σ 的函数的图。

题 解

由于在这个题目中的对称性,这个容量可以对均匀输入分布$\left(\text{即 } p(X=A)=p(X=-A)=\frac{1}{2}\right)$得到。对于这个输入分布,输出分布由下式给出:

$$p(y)=\frac{1}{2\sqrt{2\pi\sigma^2}}\mathrm{e}^{-(y+A)^2/(2\sigma^2)}+\frac{1}{2\sqrt{2\pi\sigma^2}}\mathrm{e}^{-(y-A)^2/(2\sigma^2)} \tag{8.2.12}$$

输入和输出之间的互信息为:

$$\begin{aligned}I(X;Y)=&\frac{1}{2}\int_{-\infty}^{\infty}p(y\mid X=A)\log_2\frac{p(y\mid X=A)}{p(y)}\mathrm{d}y+\\&\frac{1}{2}\int_{-\infty}^{\infty}p(y\mid X=-A)\log_2\frac{p(y\mid X=-A)}{p(y)}\mathrm{d}y\end{aligned} \tag{8.2.13}$$

用简单的积分和变量变换可得:

$$I(X;Y)=\frac{1}{2}f\left(\frac{A}{\sigma}\right)+\frac{1}{2}f\left(-\frac{A}{\sigma}\right) \tag{8.2.14}$$

其中

$$f(a)=\int_{-\infty}^{\infty}\frac{1}{\sqrt{2\pi}}\mathrm{e}^{-(u-a)^2/2}\log_2\frac{2}{1+\mathrm{e}^{-2au}}\mathrm{d}u \tag{8.2.15}$$

利用这些关系,就能对各种不同的 A/σ 值计算 $I(X;Y)$并画出结果。图 8.7 给出了所得到的曲线。

本题的 MATLAB 脚本如下所示。

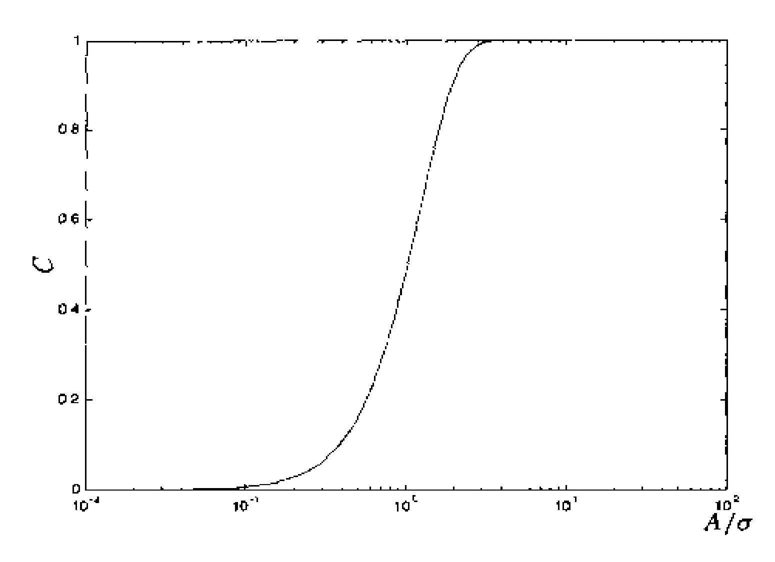


图 8.7 作为 SNR = A/σ 的函数的某个二进制输入 AWGN 信道的容量

m 文件

```
% MATLAB script for Illustrative Problem 8.3.
echo on
a_db=[-20:0.2:20];
a=10.^(a_db/10);
for i=1:201
  f(i)=quad('il3_8fun',a(i)-5,a(i)+5,1e-3,[],a(i));
  g(i)=quad('il3_8fun',-a(i)-5,-a(i)+5,1e-3,[],-a(i));
  c(i)=0.5*f(i)+0.5*g(i);
  echo off ;
end
echo on ;
pause % Press a key to see capacity vs. SNR plot.
semilogx(a,c)
title('Capacity versus SNR in binary input AWGN channel')
xlabel('SNR')
ylabel('Capacity (bits/transmission)')
```

解说题

解说题 8.4 [硬判决和软判决方法的比较]

某二进制输入信道使用了两个输入电平 A 和 $-A$。信道输出是输入和零均值、方差为 σ^2 的加性高斯白噪声之和。这个信道在两种不同的条件下应用。一种是直接用输出而不进行量化(软判定);另一种是对每个输入电平做最佳判定(硬判定)。画图示意每种情况下作为 A/σ 的函数的容量。

题 解

软判决部分与解说题 8.3 类似。对于硬判决的情况,所得二进制对称信道的交叉概率是 $Q(A/\sigma)$,因此容量为:

$$C_H = 1 - H_b\left(Q\left(\frac{A}{\sigma}\right)\right)$$

C_H 和 C_S 都如图 8.8 所示。在全部 A/σ 上,软判决解码优于硬判决解码。

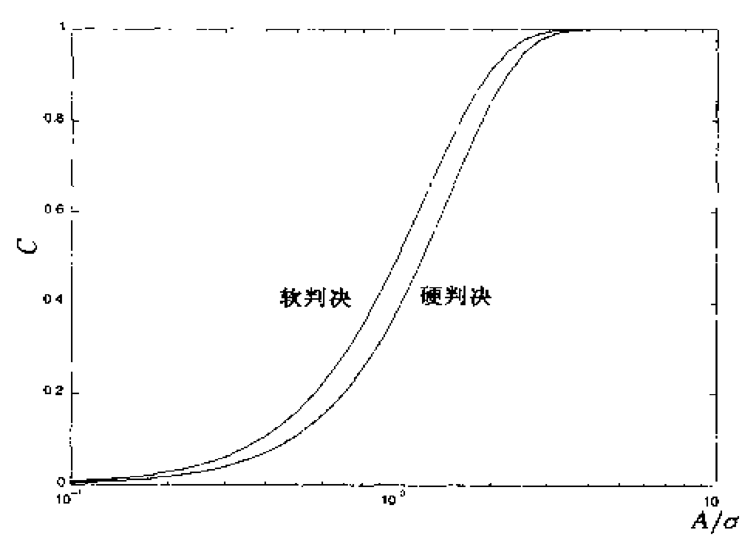


图 8.8 C_H 和 C_S 与 SNR = A/σ 的关系图

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 8.4.
echo on
a_db=[-13:0.5:13];
a=10.^(a_db/10);
c_hard=1-entropy2(Q(a));
for i=1:53
  f(i)=quad('il3_8fun',a(i)-5,a(i)+5,1e-3,[],a(i));
  g(i)=quad('il3_8fun',-a(i)-5,-a(i)+5,1e-3,[],-a(i));
  c_soft(i)=0.5*f(i)+0.5*g(i);
  echo off ;
end
echo on ;
pause % Press a key to see the capacity curves.
semilogx(a,c_soft,a,c_hard)
```

解说题

解说题 8.5 [容量与带宽和 SNR 的关系]

输入功率受限为 P 并且带宽为 W 的带限 AWGN 信道的容量由下式给出:

$$C = W\log_2\left(1 + \frac{P}{N_0 W}\right)$$

画图示意作为 W 和 P/N_0 的函数的容量。

题 解

要得到的图见图 8.9。注意,当 P/N_0 为常数时,该图就退化成图 8.6 所示的曲线;当 W 为常数时,作为 P/N_0 的函数的容量的图类似于图 8.5 所示的曲线。

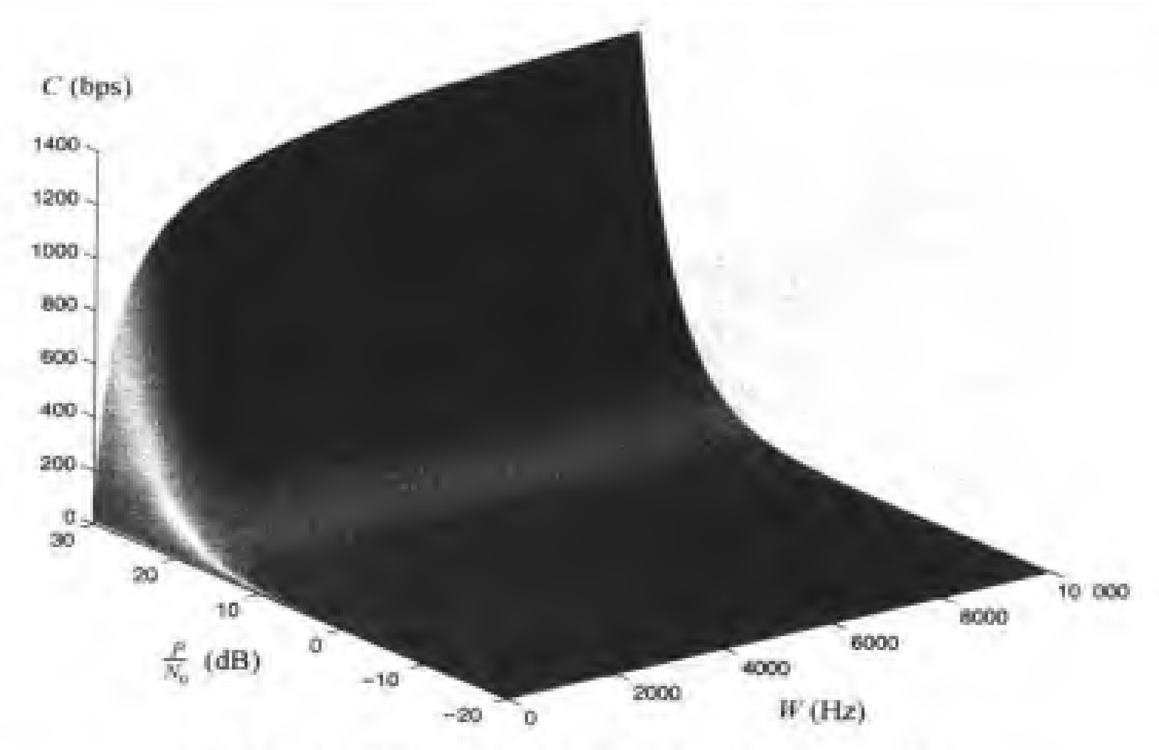


图 8.9 在 AWGN 信道中,作为带宽 W 和 SNR 的函数的容量

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 8.5.
echo off
w=[1:5:20,25:20:100,130:50:300,400:100:1000,1250:250:5000,5500:500:10000];
pn0_db=[-20:1:30];
pn0=10.^(pn0_db/10);
for i=1:45
  for j=1:51
    c(i,j)=w(i)*log2(1+pn0(j)/w(i));
  end
end
echo on
pause % Press a key to see C vs. W and P/N0.
k=[0.9,0.8,0.5,0.6];
s=[-70,35];
surfl(w,pn0_db,c',s,k)
title('Capacity vs. bandwidth and SNR')
```

解说题

解说题 8.6 [离散时间 AWGN 信道容量]

画图示意作为输入功率和噪声方差的函数的离散时间 AWGN 信道容量。

题 解

要得到的图见图 8.10。

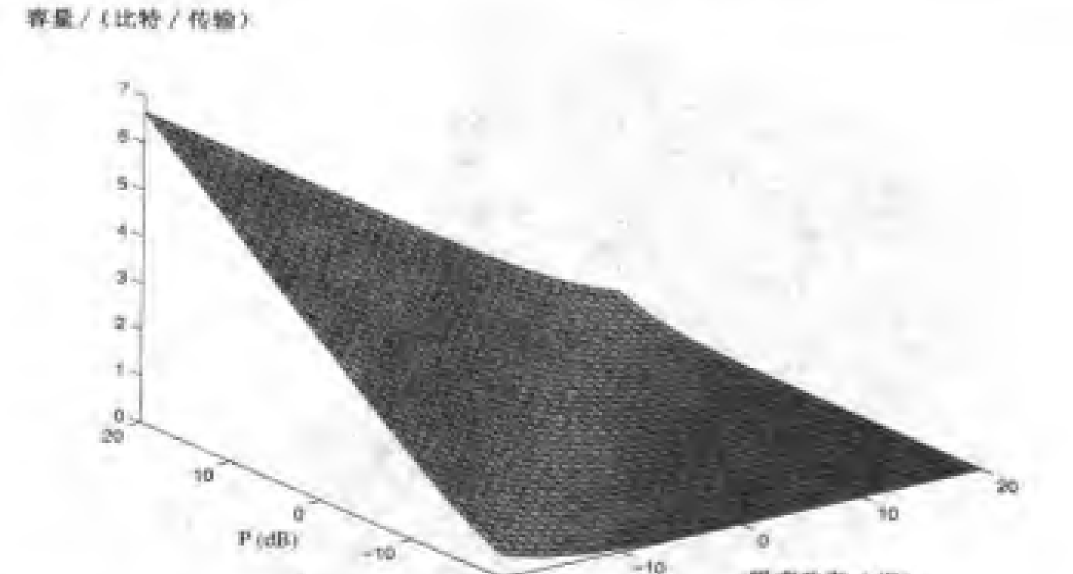


图 8.10 作为信号功率(P)和噪声功率(σ^2)的函数的离散时间 AWGN 信道容量

本题的 MATLAB 脚本如下所示。

m文件

```
% MATLAB script for Illustrative Problem 8.6.
echo on
p_db=[-20:1:20];
np_db=p_db;
p=10.^(p_db/10);
np=p;
for i=1:41
  for j=1:41
    c(i,j)=0.5*log2(1+p(i)/np(j));
    echo off ;
  end
end
echo on ;
pause % Press a key to see the plot.
surfl(np_db,p_db,c)
```

8.3 信道编码

经由有噪声信道的通信容易产生差错。为了减少差错的影响并实现可靠的通信,有必要使传输的序列尽可能地不同,以使信道噪声不会将一个序列变成另一个序列。这就意味着不得不引入某些冗余度,以提高通信的可靠性。冗余度的引入会导致额外的比特传输,从而降低传输速率。

一般可以将信道编码方法分成两大类:分组码和卷积码。在分组编码中,将长度为 k 的二进制信源输出序列映射为长度为 n 的二进制信源输入序列;因此,所得到的码率就是每次传输 k/n 比特。这样的一种码称为(n,k)分组码,并由长度为 n 的 2^k 个码字组成,通常记为 $c_1,c_2,\cdots,c_{2^k}$。信源输出映射到信道输入是独立完成的,而编码器输出仅取决于当前长度为 k 的输入序

列，与先前的输入序列无关。在卷积编码中，长度为 k_0 的信源输出映射为 n_0 的信道输入，但是该信道输入不仅与最近的 k_0 个信源输出有关，还与前面的 $(L-1)k_0$ 个编码器输入有关。

最简单的一种分组码是**简单重复码**(simple repetition code)，其中有两个消息要在一个二进制对称信道上传输，对这两个消息不用 0 和 1 传输，而是用由全 0 和全 1 组成的两个序列来传输。这两个序列的长度选为某个奇数 n，其编码过程如下所示：

$$0 \to \overbrace{00 \cdots 00}^{n\text{为奇数}} \tag{8.3.1}$$

$$1 \to \overbrace{11 \cdots 11}^{n\text{为奇数}} \tag{8.3.2}$$

解码是按照“简单多数表决”的方式进行的；也就是说，如果接收符号大多数是 1，解码器判决为 1；如果大多数是 0，解码器判决为 0。

如果接收到的传输符号至少有 $(n+1)/2$ 个是错误的，那么就会发生差错。因为信道是具有交叉概率为 ϵ 的二进制对称信道，差错概率可以表示为：

$$p_e = \sum_{k=(n+1)/2}^{n} \binom{n}{k} \epsilon^k (1-\epsilon)^{n-k} \tag{8.3.3}$$

例如，当 $n=5, \epsilon=0.001$ 时有：

$$p_e = \sum_{k=3}^{5} 0.001^k (0.999)^{5-k} = 9.99 \times 10^{-10} \approx 10^{-9} \tag{8.3.4}$$

这就是说，利用信道 5 次而不是 1 次就能将差错概率从 0.001 降低到 10^{-9}！当然，为这个更可靠的性能已经付出了代价，这个代价就是降低了传输速率并增加了系统的复杂性。传输速率从每使用信道 1 次传输一个二进制消息降低到每使用信道 5 次才传输一个二进制消息。由于现在必须用一个编码器(结构很简单)和一个解码器(用于实现简单多数表决解码)，所以系统复杂性也就增加了。在这个解说题中，如果继续增加 n 还能实现更可靠的通信。例如，对 $n=9$ 有：

$$p_e = \sum_{k=5}^{9} 0.001^k (0.999)^{9-k} = 9.97 \times 10^{-16} \approx 10^{-15} \tag{8.3.5}$$

由上面的讨论似乎可以得出，若想将差错概率减到零，就要无限增加 n，因此传输速率降到 0！然而，情况并非是这样，香农指出，只要将传输速率保持在信道容量以下，就能渐近地实现可靠的通信(即 $p_e \to 0$)。在上述情况下，这就是：

$$C = 1 - H_b(0.001) = 1 - 0.0114 = 0.9886 \quad (\text{比特/传输}) \tag{8.3.6}$$

然而，这是在通过使用编码和解码方法之后才能达到的，而这些编码和解码比简单重复编码复杂得多。

解说题

解说题 8.7　[简单重复编码中的差错概率]

假设在某个二进制对称信道中 $\epsilon=0.3$，画图示意作为分组长度 n 的函数的 p_e。

题　解

差错概率给出为：

$$p_e = \sum_{k=(n+1)/2}^{n} \binom{n}{k} 0.3^k \times 0.7^{n-k}$$

对于 n 从1变化到61,导出 p_e。图8.11是所得到的图。

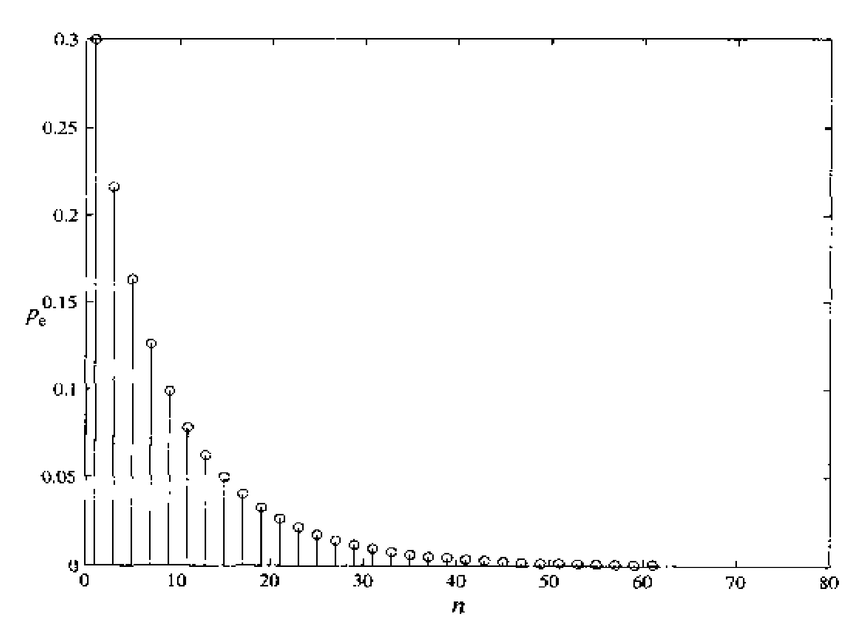


图8.11 在 $\varepsilon=0.3$ 和 $n=1,2,3,\cdots,61$ 时简单重复编码的差错概率

本题的MATLAB脚本如下所示。

m文件

```
% MATLAB script for Illustrative Problem 8.7
echo on
ep=0.3;
for i=1:2:61
  p(i)=0;
  for j=(i+1)/2:i
    p(i)=p(i)+prod(1:i)/(prod(1:j)*prod(1:(i-j)))*ep^j*(1-ep)^(i-j);
    echo off ;
  end
end
echo on ;
pause % Press a key to see the plot.
stem((1:2:61),p(1:2:61))
xlabel('n')
ylabel('pe')
title('Error probability as a function of n in simple repetition code')
```

8.3.1 线性分组码

线性分组码是最重要和应用最广泛的一类分组码。如果任意两个码字的线性组合还是一个码字,这种分组码就是线性的。在二进制情况下,这就意味着任意两个码字的和还是一个码字。在线性分组码中,码字形成 n 维空间的一个 k 维子空间。线性分组码是利用它们的**生成矩阵**(generator matrix) $\boldsymbol{G}$ 来描述的,它是一个 $k\times n$ 的二进制矩阵,以使每个码字 $\boldsymbol{c}$ 都能写成如下形式:

$$c = uG \tag{8.3.7}$$

其中，u 是长度为 k 的二进制数据序列（编码器输入）。很显然，长度为 n 的全 0 序列总是 (n,k) 线性分组码中一个码字。

在线性分组码中，一个重要的参数是码的最小 Hamming 距离，它决定了该码的误差校正能力，定义为任意两个不同码字之间的最小 Hamming 距离。一个码的最小距离用 $d_{\min}$ 表示为：

$$d_{\min} = \min_{i \neq j} d_{\mathrm{H}}(\boldsymbol{c}_i, \boldsymbol{c}_j) \tag{8.3.8}$$

对于线性码，这个最小距离等于该码的最小重量，定义为：

$$w_{\min} = \min_{\boldsymbol{c}_i \neq \boldsymbol{0}} w(\boldsymbol{c}_i) \tag{8.3.9}$$

也就是说，在任何非零码字中 1 的个数最少。

解说题

解说题 8.8　[线性分组码]

一个(10,4)线性分组码的生成矩阵为：

$$\boldsymbol{G} = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}$$

求全部码字和该码的最小重量。

题　解

为了求得全部码字，必须用到长度为 4 的全部信息序列，并找出对应的编码序列。因为总共有 16 个长度为 4 的二进制序列，所以将有 16 个码字。令 $\boldsymbol{U}$ 记为 $2^k \times k$ 的矩阵，该矩阵的行是长为 k 的所有可能的二进制序列，由全部为 0 的序列开始，并以全部为 1 的序列结束。各行按下述方法排列：按序列的十进制大小，从上至下由小到大排列。对于 $k=4$ 的情况，矩阵 $\boldsymbol{U}$ 为：

$$\boldsymbol{U} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

这样有：

$$C = UG \tag{8.3.10}$$

其中，C 是码字矩阵，在这个例子中是 16×10 的矩阵，它的行是码字。这个码字矩阵为：

$$C = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

仔细检查这些码字指出，这个码字的最小距离是 $d_{min} = 2$。

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 8.8.
% Generate U, denoting all information sequences.
k=4;
for i=1:2^k
  for j=k:-1:1
    if rem(i-1,2^(-j+k+1))>=2^(-j+k)
      u(i,j)=1;
    else
      u(i,j)=0;
    end
    echo off ;
  end
end
echo on ;
% Define G, the generator matrix.
g=[1 0 0 1 1 1 0 1 1 1;
   1 1 1 0 0 0 1 1 1 0;
   0 1 1 0 1 1 0 1 0 1;
   1 1 0 1 1 1 1 0 0 1];
% Generate codewords.
c=rem(u*g,2);
% Find the minimum distance.
w_min=min(sum((c(2:2^k,:))'));
```

如果生成矩阵是下列形式：

$$G=\begin{bmatrix} 1 & 0 & \cdots & 0 & p_{1,1} & p_{1,2} & \cdots & p_{1,n-k} \\ 0 & 1 & \cdots & 0 & p_{2,1} & p_{2,2} & \cdots & p_{2,n-k} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & p_{k,1} & p_{k,2} & \cdots & p_{k,n-k} \end{bmatrix} \tag{8.3.11}$$

或者

$$G=[I_k \mid P] \tag{8.3.12}$$

那么线性分组码就具有**规则形式**(systematic form)，其中，I_k 代表 $k\times k$ 的单位矩阵，而 P 代表 $k\times(n-k)$的矩阵。在一个规则码中，一个码字的前 k 个二进制符号是信息比特，而余下的 $n-k$个二进制符号是**奇偶校验符号**(parity-check symbol)。

一个码字的**奇偶校验矩阵**(parity-check matrix)是任意$(n-k)\times n$ 的二进制矩阵 H，使得对全部码字 c 有：

$$cH^{t}=0 \tag{8.3.13}$$

显然

$$GH^{t}=0 \tag{8.3.14}$$

并且，如果 G 是规则形式，则有：

$$H=[P^{t} \mid I^{k}] \tag{8.3.15}$$

Hamming 码

Hamming 码是$(2^m-1,2^m-m-1)$的线性分组码，其最小距离为 3，且有一个很简单的奇偶

校验矩阵。奇偶校验矩阵是 $m\times(2^m-1)$的矩阵,除去全0序列外,用长为 m 的全部序列作为它的列。例如,对于 $m=3$,有一个(7,4)的码,它的奇偶校验矩阵用规则形式表示是:

$$\boldsymbol{H}=\begin{bmatrix}1&0&1&1&1&0&0\\1&1&0&1&0&1&0\\0&1&1&1&0&0&1\end{bmatrix}\tag{8.3.16}$$

据此有:

$$\boldsymbol{G}=\begin{bmatrix}1&0&0&0&1&1&0\\0&1&0&0&0&1&1\\0&0&1&0&1&0&1\\0&0&0&1&1&1&1\end{bmatrix}\tag{8.3.17}$$

解说题

解说题 8.9 [Hamming 码]

求(15,11)的 Hamming 码的全部码字,并验证它的最小距离等于3。

题 解

现在

$$\boldsymbol{H}=\begin{bmatrix}1&0&0&1&1&0&1&0&1&1&1&1&0&0&0\\1&1&0&0&0&1&1&1&0&1&1&0&1&0&0\\0&1&1&1&0&0&1&1&1&0&1&0&0&1&0\\0&0&1&0&1&1&0&1&1&1&1&0&0&0&1\end{bmatrix}\tag{8.3.18}$$

因此有:

$$\boldsymbol{G}=\begin{bmatrix}1&0&0&0&0&0&0&0&0&0&0&1&1&0&0\\0&1&0&0&0&0&0&0&0&0&0&0&1&1&0\\0&0&1&0&0&0&0&0&0&0&0&0&0&1&1\\0&0&0&1&0&0&0&0&0&0&0&1&0&1&0\\0&0&0&0&1&0&0&0&0&0&0&1&0&0&1\\0&0&0&0&0&1&0&0&0&0&0&0&1&0&1\\0&0&0&0&0&0&1&0&0&0&0&1&1&1&0\\0&0&0&0&0&0&0&1&0&0&0&0&1&1&1\\0&0&0&0&0&0&0&0&1&0&0&1&0&1&1\\0&0&0&0&0&0&0&0&0&1&0&1&1&0&1\\0&0&0&0&0&0&0&0&0&0&1&1&1&1&1\end{bmatrix}$$

总共有 $2^{11}=2048$ 个码字,每个都是15位长。码率是$\frac{11}{15}=0.733$。为了验证该码的最小距离,我们用类似于在解说题8.8中所用的 MATLAB 脚本。这个 MATLAB 脚本如下所示,其结果 $d_{\min}=3$。

m 文件

```
% MATLAB script for Illustrative Problem 8.9.
echo on
k=11;
```

```
for i=1:2^k
  for j=k:-1:1
    if rem(i-1,2^(-j+k+1))>=2^(-j+k)
      u(i,j)=1;
    else
      u(i,j)=0;
    end
    echo off ;
  end
end
echo on ;

g=[1 0 0 0 0 0 0 0 0 0 0 1 1 0 0;
   0 1 0 0 0 0 0 0 0 0 0 0 1 1 0;
   0 0 1 0 0 0 0 0 0 0 0 0 0 1 1;
   0 0 0 1 0 0 0 0 0 0 0 1 0 1 0;
   0 0 0 0 1 0 0 0 0 0 0 1 0 0 1;
   0 0 0 0 0 1 0 0 0 0 0 0 1 0 1;
   0 0 0 0 0 0 1 0 0 0 0 1 1 1 0;
   0 0 0 0 0 0 0 1 0 0 0 0 1 1 1;
   0 0 0 0 0 0 0 0 1 0 0 1 0 1 1;
   0 0 0 0 0 0 0 0 0 1 0 1 1 0 1;
   0 0 0 0 0 0 0 0 0 0 1 1 1 1 1];

c=rem(u*g,2);
w_min=min(sum((c(2:2^k,:))'));
```

线性分组码的性能

对线性分组码进行的解码既可以用软判决解码,也可以用硬判决解码。在硬判决解码方法中,首先在码元上按位进行判决,然后用最小 Hamming 距离准则完成解码。这种解码方法的性能决定于码的距离结构,但还是能够利用码的最小距离获得一种严格的上界的(特别是在高 SNR 值的情况下)。

最小距离为 d_{min} 的线性分组码,在硬判决解码时它的(消息)差错概率的上界为:

$$p_e \leqslant (M-1)[4p(1-p)]^{d_{min}/2} \tag{8.3.19}$$

其中,p 代表二进制信道的差错概率(解调中的差错概率),M 是码字的数目($M=2^k$)。

在软判决解码中,将接收信号映射到一个码字,该码字对应的信号与接收信号有最小的欧氏距离。在这种情况下,消息差错概率的上界为:

$$p_e \leqslant (M-1)Q\left(\frac{d^E}{\sqrt{2N_0}}\right) \tag{8.3.20}$$

其中,$M=2^k$ 是码字的数目,N_0 是单边噪声功率谱密度,而 d^E 是该码的**最小欧氏距离**,由下式给出:

$$d^E = \begin{cases} \sqrt{2d_{min}\mathcal{E}}, & \text{对正交信号} \\ \sqrt{4d_{min}\mathcal{E}}, & \text{对反极性信号} \end{cases} \tag{8.3.21}$$

这样可得:

$$p_e \leqslant \begin{cases} (M-1)Q\left(\sqrt{\dfrac{d_{\min}\mathcal{E}}{N_0}}\right), & \text{对正交信号} \\ (M-1)Q\left(\sqrt{\dfrac{2d_{\min}\mathcal{E}}{N_0}}\right), & \text{对反极性信号} \end{cases} \tag{8.3.22}$$

在这些不等式中,$d_{\min}$ 是码的最小 Hamming 距离,$\mathcal{E}$ 表示码字中每个码元的能量。因为每个码字有 n 个码元,所以每个码字的能量是 $n\mathcal{E}$;又因为每个码字载有 k 个信息比特,所以每比特的能量 $\mathcal{E}_b$ 为:

$$\mathcal{E}_b = \frac{n\mathcal{E}}{k} = \frac{\mathcal{E}}{R_c} \tag{8.3.23}$$

其中,$R_c = k/n$ 代表码率。因此,上面的关系可写为:

$$p_e \leqslant \begin{cases} (M-1)Q\left(\sqrt{\dfrac{d_{\min}R_c\mathcal{E}_b}{N_0}}\right), & \text{对正交信号} \\ (M-1)Q\left(\sqrt{\dfrac{2d_{\min}R_c\mathcal{E}_b}{N_0}}\right), & \text{对反极性信号} \end{cases} \tag{8.3.24}$$

所得到的这个上界通常仅对大的 $\gamma_b = \mathcal{E}_b/N_0$ 值有用。对于较小的 γ_b 值,这个上界就变得不严格了,甚至还可能超过 1。

解说题

解说题 8.10 [硬判决解码的性能]

假设(15,11)的 Hamming 码用于反极性信号和硬判决解码,画出作为 $\gamma_b = \mathcal{E}_b/N_0$ 的函数的消息差错概率。

题 解

因为使用的是反极性信号,所以二进制信道的差错概率为:

$$p = Q\left(\sqrt{\frac{2\mathcal{E}}{N_0}}\right) \tag{8.3.25}$$

其中,$\mathcal{E}$ 是每个码元的能量(每维的能量),它由 $\mathcal{E}_b$ 导出为:

$$\mathcal{E} = \mathcal{E}_b R_c \tag{8.3.26}$$

因此,

$$p = Q\left(\sqrt{\frac{2R_c\mathcal{E}_b}{N_0}}\right) \tag{8.3.27}$$

其中,$R_c = k/n = \dfrac{11}{15} = 0.733\,33$。因为 Hamming 码的最小距离是 3,所以有:

$$\begin{aligned} p_e &\leqslant (2^{11}-1)[4p(1-p)]^{d_{\min}/2} \\ &= 2047\left(4Q\left(\sqrt{\frac{1.466\mathcal{E}_b}{N_0}}\right)\left(1-Q\left(\sqrt{\frac{1.466\mathcal{E}_b}{N_0}}\right)\right)\right]^{1.5} \end{aligned} \tag{8.3.28}$$

图 8.12 是所得的关系图。

当采用硬判决和反极性信号时,计算线性分组码的消息差错概率上界的 MATLAB 函数如下所示。

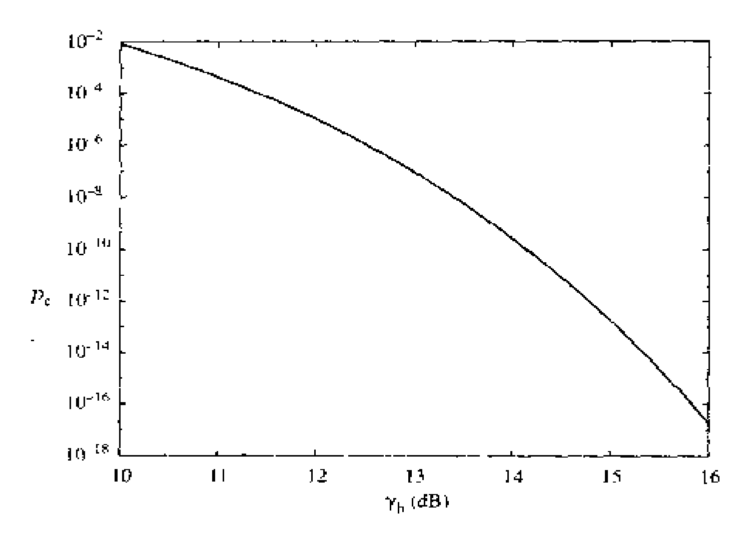


图 8.12　采用硬判决解码和反极性信号时,对于(15,11)的 Hamming 码,作为 γ_b 的函数的差错概率

m 文件

```
function [p_err,gamma_db]=p_e_hd_a(gamma_db_l,gamma_db_h,k,n,d_min)
% p_e_hd_a.m      Matlab function for computing error probability in
%                 hard-decision decoding of a linear block code
%                 when antipodal signaling is used.
%                 [p_err,gamma_db]=p_e_hd_a(gamma_db_l,gamma_db_h,k,n,d_min)
%                 gamma_db_l=lower E_b/N_0
%                 gamma_db_h=higher E_b/N_0
%                 k=number of information bits in the code
%                 n=code block length
%                 d_min=minimum distance of the code

gamma_db=[gamma_db_l:(gamma_db_h-gamma_db_l)/20:gamma_db_h];
gamma_b=10.^(gamma_db/10);
R_c=k/n;
p_b=q(sqrt(2.*R_c.*gamma_b));
p_err=(2^k-1).*(4*p_b.*(1-p_b)).^(d_min/2);
```

在下面给出的 MATLAB 脚本中,用前面的 MATLAB 函数画出了差错概率与 γ_b 的关系图。

m 文件

```
% MATLAB script for Illustrative Problem 8.10.
[p_err_ha,gamma_b]=p_e_hd_a(10,16,11,15,3);
semilogy(gamma_b,p_err_ha)
```

解说题

解说题 8.11　[硬判决解码]

如果(15,11)的 Hamming 码不使用反极性方法,而是使用正交二进制调制方法,画出作为 $\gamma_b = \mathcal{E}_b/N_0$ 的函数的消息差错概率。

题　解

除了该等效二进制对称信道的交叉概率(硬判决解码之后)由下式:

$$p = Q\left(\sqrt{\frac{\mathcal{E}}{N_0}}\right) \tag{8.3.29}$$

给出以外,这个题与解说题 8.10 是类似的。利用关系:

$$\mathcal{E} = \mathcal{E}_b R_c \tag{8.3.30}$$

可得:

$$p = Q\left(\sqrt{\frac{R_c \mathcal{E}_b}{N_0}}\right) \tag{8.3.31}$$

最后有

$$\begin{aligned} p_e &\leqslant (2^{11}-1)[4p(1-p)]^{d_{\min}/2} \\ &= 2047\left[4Q\left(\sqrt{\frac{0.733\,\mathcal{E}_b}{N_0}}\right)\left(1-Q\left(\sqrt{\frac{0.733\,\mathcal{E}_b}{N_0}}\right)\right)\right]^{1.5} \end{aligned} \tag{8.3.32}$$

$\gamma_e \sim \mathcal{E}_b/N_0$ 的图见图 8.13。

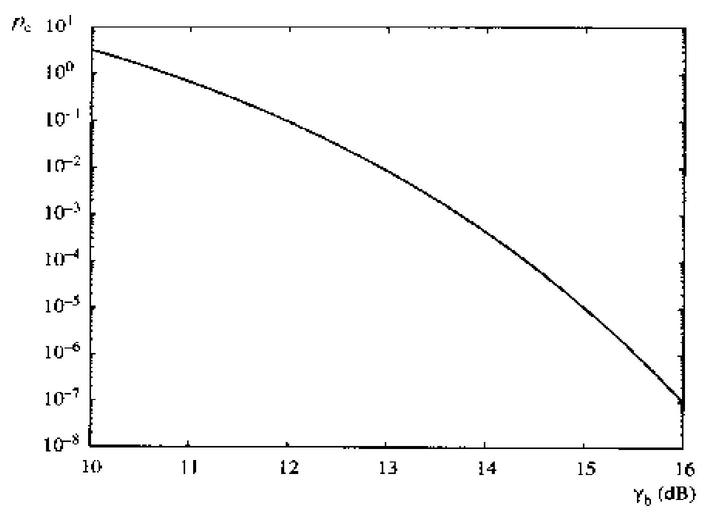


图 8.13　使用正交信号和硬判决解码时,(15,11)码的差错概率与 γ_b 的关系图

本题的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 8.11.
echo on
gamma_b_db=[-4:1:14];
gamma_b=10.^(gamma_b_db/10);
qq=q(sqrt(0.733.*gamma_b));
p_err=2047*qq.^2.*(3-2.*qq);
pause % Press a key to see p_err versus gamma_b curve.
loglog(gamma_b,p_err)
```

正如在图 8.13 中所看到的,对于较低的 γ_b 值,所导得的上界就不太严格了。事实上,对这些 γ_b 值,差错概率的上界是大于 1 的。将正交信号和反极性信号这两个差错概率的上界画

在同一张图上也是有启发意义的，如图 8.14 所示。从这两个图的比较中很容易看出，反极性信号与正交信号相比具有更优越的性能。

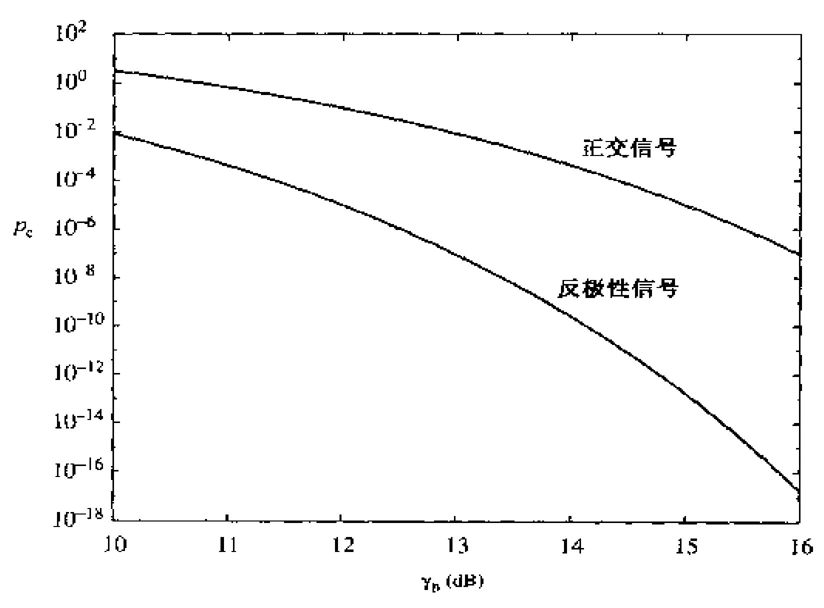


图 8.14　反极性信号与正交信号的比较

在硬判决解码情况下，下面的 MATLAB 函数可以用于计算用正交信号表示的消息的差错概率。

m 文件

```
function [p_err,gamma_db]=p_e_hd_o(gamma_db_l,gamma_db_h,k,n,d_min)
% p_e_hd_o.m     Matlab function for computing error probability in
%                hard-decision decoding of a linear block code
%                when orthogonal signaling is used.
%                [p_err,gamma_db]=p_e_hd_o(gamma_db_l,gamma_db_h,k,n,d_min)
%                gamma_db_l=lower E_b/N_0
%                gamma_db_h=higher E_b/N_0
%                k=number of information bits in the code
%                n=code block length
%                d_min=minimum distance of the code

gamma_db=[gamma_db_l:(gamma_db_h-gamma_db_l)/20:gamma_db_h];
gamma_b=10.^(gamma_db/10);
R_c=k/n;
p_b=q(sqrt(R_c.*gamma_b));
p_err=(2^k-1).*(4*p_b.*(1-p_b)).^(d_min/2);
```

解说题

解说题 8.12　[软判决解码]

当使用软判决解码而不使用硬判决解码时，重做解说题 8.11。

题　解

在这种情况下，必须用式(8.3.24)求差错概率的上界。在本题中，$d_{\min}=3$，$R_c=\frac{11}{15}$，$M=2^{11}-1=2047$。因此有：

$$p_e \leqslant \begin{cases} (M-1)Q\left(\sqrt{\dfrac{d_{\min}R_c\mathcal{E}_b}{N_0}}\right), & \text{对正交信号} \\ (M-1)Q\left(\sqrt{\dfrac{2d_{\min}R_c\mathcal{E}_b}{N_0}}\right), & \text{对反极性信号} \end{cases}$$

$$\leqslant \begin{cases} 2047\,Q\left(\sqrt{\dfrac{11}{5}\dfrac{\mathcal{E}_b}{N_0}}\right), & \text{对正交信号} \\ 2047\,Q\left(\sqrt{\dfrac{22}{5}\dfrac{\mathcal{E}_b}{N_0}}\right), & \text{对反极性信号} \end{cases}$$

对应的图见图 8.15。由该图可见,反极性信号的优越性能是显而易见的。

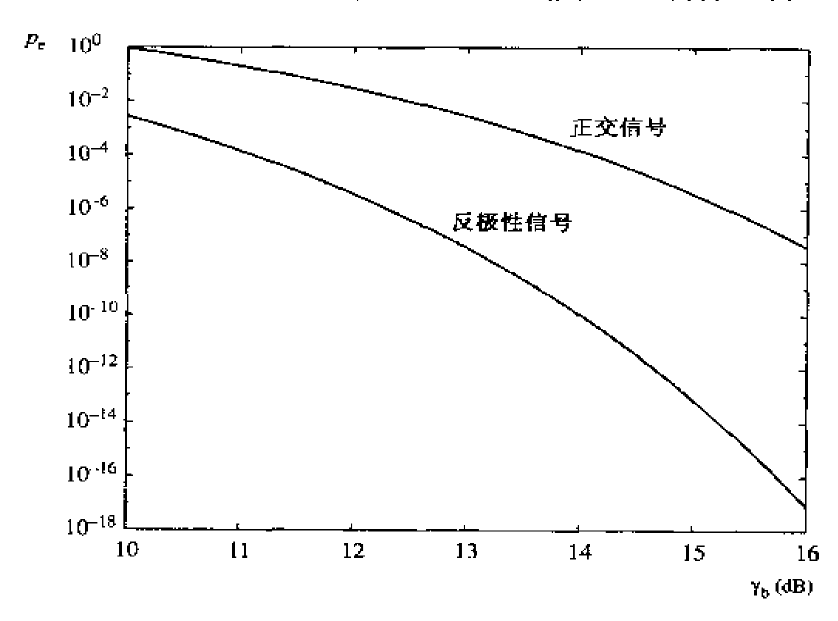


图 8.15　使用软判决解码时,消息差错概率与 γ_b 的关系图

下面给出两个 MATLAB 函数,一个针对反极性信号计算差错概率,另一个针对正交信号计算差错概率,两者都使用了软判决解码。

m 文件

```
function [p_err,gamma_db]=p_e_sd_a(gamma_db_l,gamma_db_h,k,n,d_min)
% p_e_sd_a.m     Matlab function for computing error probability in
%                soft-decision decoding of a linear block code
%                when antipodal signaling is used.
%                [p_err,gamma_db]=p_e_sd_a(gamma_db_l,gamma_db_h,k,n,d_min)
%                gamma_db_l=lower E_b/N_0
%                gamma_db_h=higher E_b/N_0
%                k=number of information bits in the code
%                n=code block length
%                d_min=minimum distance of the code

gamma_db=[gamma_db_l:(gamma_db_h-gamma_db_l)/20:gamma_db_h];
gamma_b=10.^(gamma_db/10);
R_c=k/n;
p_err=(2^k-1).*q(sqrt(2.*d_min.*R_c.*gamma_b));
```

m 文件

```
function [p_err,gamma_db]=p_e_sd_o(gamma_db_l,gamma_db_h,k,n,d_min)
% p_e_sd_o.m      Matlab function for computing error probability in
%                 soft-decision decoding of a linear block code
%                 when orthogonal signaling is used.
%                 [p_err,gamma_db]=p_e_sd_o(gamma_db_l,gamma_db_h,k,n,d_min)
%                 gamma_db_l=lower E_b/N_0
%                 gamma_db_h=higher E_b/N_0
%                 k=number of information bits in the code
%                 n=code block length
%                 d_min=minimum distance of the code

gamma_db=[gamma_db_l:(gamma_db_h-gamma_db_l)/20:gamma_db_h];
gamma_b=10.^(gamma_db/10);
R_c=k/n;
p_err=(2^k-1).*q(sqrt(d_min.*R_c.*gamma_b));
```

图 8.16 给出了 4 个图，分别对应于用软判决(SD)和硬判决(HD)时，采用反极性信号和正交信号的情况。

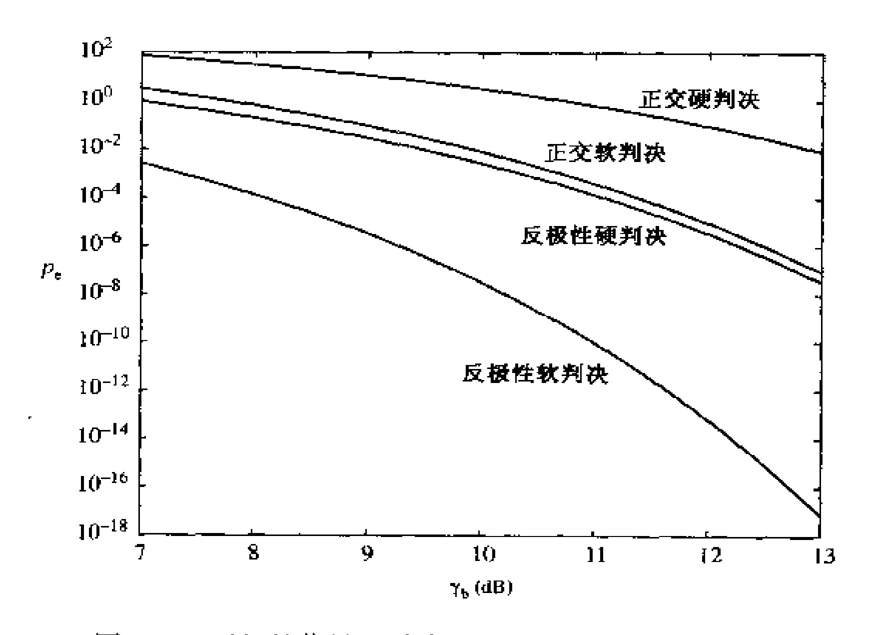


图 8.16　反极性信号/正交信号和软/硬判决解码的对比

产生图 8.16 的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 8.12.
[p_err_ha,gamma_b]=p_e_hd_a(7,13,11,15,3);
[p_err_ho,gamma_b]=p_e_hd_o(7,13,11,15,3);
[p_err_so,gamma_b]=p_e_sd_o(7,13,11,15,3);
[p_err_sa,gamma_b]=p_e_sd_a(7,13,11,15,3);
semilogy(gamma_b,p_err_sa,gamma_b,p_err_so,gamma_b,p_err_ha,gamma_b,p_err_ho)
```

8.3.2 卷积码

在分组码中,每 k 信息比特的序列以某种固定的方式映射为一个 n 信道输入的序列,但是与前面的信息比特无关。在卷积码中,每 k_0 个信息比特序列映射为一个长为 n_0 的信道输入序列,但是这个信道输出序列不仅取决于最当前的 k_0 个信息比特,而且还与该编码器的前 $(L-1)k_0$ 个输入有关。因此,这种编码器有一种有限状态机的结构,在其中的每一时刻,输出序列不但与输入序列有关,而且与编码器的状态有关,这个状态是由编码器的前 $(L-1)k_0$ 个输入决定的。参数 L 称为该卷积码的约束长度①。因此,一个二进制卷积码就是一个具有 $2^{k_0(L-1)}$ 个状态的有限状态机。图 8.17 给出了 $k_0=2, n_0=3$ 和 $L=4$ 的一种卷积码的方框图。

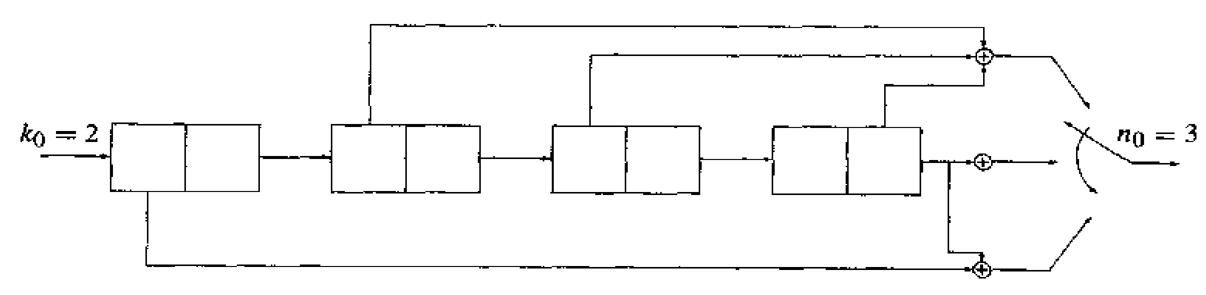


图 8.17　$k_0=2, n_0=3$ 和 $L=4$ 的一种卷积码

在这个卷积编码器中,信息比特是通过每次 2 比特送入这个移位寄存器的,而在该移位寄存器中的最后 2 个信息比特移出。这 3 个编码比特然后按图示计算出并在信道上传输。因此,这个码的速率是 $R=2/3$。注意,在信道上传输的这 3 个编码器输出既决定于输入到这个移位寄存器的 2 个信息比特,也与该移位寄存器的前三级(6 比特)内容有关。最后一级(2 比特)的内容在输出上没有影响,因为一旦 2 信息比特输入进去,它们就从该移位寄存器出去了。

通常,我们用卷积码的**生成序列**(generator sequence)来定义卷积码,记为 $\boldsymbol{g}_1, \boldsymbol{g}_2, \cdots, \boldsymbol{g}_n$。如果这个移位寄存器的第 i 个单元连到对应于输出的第 j 个比特的相加器上,那么 $\boldsymbol{g}_j$ 中的第 i 个元素就是 1,否则就是 0,其中 $1 \leqslant i \leqslant k_0 L, 1 \leqslant j \leqslant n$。例如,在图 8.17 中表示的卷积码,其生成序列为:

$$\boldsymbol{g}_1 = [0\quad 0\quad 1\quad 0\quad 1\quad 0\quad 0\quad 1]$$
$$\boldsymbol{g}_2 = [0\quad 0\quad 0\quad 0\quad 0\quad 0\quad 0\quad 1]$$
$$\boldsymbol{g}_3 = [1\quad 0\quad 0\quad 0\quad 0\quad 0\quad 0\quad 1]$$

一旦给定了 $\boldsymbol{g}_1, \boldsymbol{g}_2, \cdots, \boldsymbol{g}_n$,卷积码就被惟一确定了。

我们还定义卷积码的生成矩阵为:

$$\boldsymbol{G} = \begin{bmatrix} \boldsymbol{g}_1 \\ \boldsymbol{g}_2 \\ \vdots \\ \boldsymbol{g}_n \end{bmatrix}$$

① 有些作者定义 $m=Lk_0$ 为约束长度,而另一些作者又喜欢定义 $(L-1)k_0$ 为约束长度。因作者而异。

一般来说,这是一个 $n \times k_0 L$ 的矩阵。对于图 8.17 所示的卷积码,其生成矩阵为:

$$G = \begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

假定产生卷积码的移位寄存器在进入第 1 个信息比特之前全都是 0 输入(即该编码器初始为 0 状态),并且假定将信息比特序列补 $(L-1)k_0$ 个 0,以便使该卷积编码器又回到 0 状态,这样做是有用的。另外,还假设信息比特序列(卷积编码器的输入)的长度是 k_0 的倍数。这个信息比特序列的长度不是 k_0 的倍数,就给它补 0,使得到的长度是 k_0 的倍数。这个要在本段先前指出的补 $(L-1)k_0$ 个 0 之前完成。如果在第 1 次补 0 之后,这个输入序列的长度是 nk_0,那么输出序列将是 $(n+L-1)n_0$,因此码率为:

$$\frac{nk_0}{(n+L-1)n_0}$$

在实际情况中,n 要比 L 大得多,所以上式就能很好地近似为:

$$R_c = \frac{k_0}{n_0}$$

下面给出的 MATLAB 函数 cnv _ encd.m 是当 G, k_0 和输入序列给定时,产生这个卷积编码器的输入序列。注意,输入序列的补零是用 MATLAB 函数完成的。输入序列(用参数 input 表示)从进入编码器的第 1 个信息比特开始。参数 n_0 和 L 由矩阵 G 导出。

m 文件

```
function output=cnv_encd(g,k0,input)
%          cnv_encd(g,k0,input)
%          determines the output sequence of a binary convolutional encoder
%          g is the generator matrix of the convolutional code
%          with n0 rows and l*k0 columns. Its rows are g1,g2,...,gn.
%          k0 is the number of bits entering the encoder at each clock cycle.
%          input the binary input seq.

% Check to see if extra zero-padding is necessary.
if rem(length(input),k0) > 0
  input=[input,zeros(size(1:k0-rem(length(input),k0)))];
end
n=length(input)/k0;
% Check the size of matrix g.
if rem(size(g,2),k0) > 0
  error('Error, g is not of the right size.')
end
% Determine l and n0.
l=size(g,2)/k0;
n0=size(g,1);
% add extra zeros
u=[zeros(size(1:(l-1)*k0)),input,zeros(size(1:(l-1)*k0))];
% Generate uu, a matrix whose columns are the contents of
% conv. encoder at various clock cycles.
u1=u(l*k0:-1:1);
for i=1:n+l-2
  u1=[u1,u((i+l)*k0:-1:i*k0+1)];
```

```
end
uu=reshape(u1,l*k0,n+l-1);
%  Determine the output.
output=reshape(rem(g*uu,2),1,n0*(l+n-1));
```

解说题

解说题 8.13 [卷积编码器]

当信息序列为

1 0 0 1 1 1 0 0 1 1 0 0 0 0 1 1 1

时,求图 8.17 所示的卷积编码器的输出。

题 解

现在该信息序列的长是 17,它不是 $k_0 = 2$ 的倍数。因此,现在补一个额外的 0 就足够了,这样长度就为 18。于是就有了下面的信息序列:

1 0 0 1 1 1 0 0 1 1 0 0 0 0 1 1 1 0

现在,因为有:

$$G = \begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

可得 $n_0 = 3$ 和 $L = 4$(由图 8.17 也显然可见)。因此,输出序列的长度是:

$$\left(\frac{18}{2} + 4 - 1\right) \times 3 = 36$$

为确保编码器从全 0 状态开始,并回到全 0 状态,要求在输入序列的起始和末尾都添加 $(L-1)k_0$ 个 0。因此,我们正在讨论的序列就变成:

0 0 0 0 0 0 1 0 0 1 1 1 0 0 1 1 0 0 0 0 1 1 1 0 0 0 0 0 0 0

利用函数 cnv_encd.m,求得输出序列为:

0 0 0 0 0 1 1 0 1 1 1 1 1 0 1 0 1 1 1 0 0 1 1 0 1 0 0 1 0 0 1 1 1 1 1 1

解这个题的 MATLAB 脚本如下所示。

m 文件

```
k0=2;
g=[0 0 1 0 1 0 0 1;0 0 0 0 0 0 0 1;1 0 0 0 0 0 0 1];
input=[1 0 0 1 1 1 0 0 1 1 0 0 0 0 1 1 1];
output=cnv_encd(g,k0,input);
```

卷积码的表示法

我们已经看到,一个卷积码既可以用编码器的结构,也可以用生成矩阵 G 来表示。另外,我们还知道一个卷积编码器能够表示成一个有限状态机,因此它就能用代表这个有限状态机的状态转移图来描述。对卷积码的表示来说,应用更广泛的方法是使用它们的**网格图**(trellis diagram)来表示。一个网格图就是一个按时间变化的状态转移图。因此,网格图是一个 $2^{(L-1)k_0}$ 个状态的

序列(对每个时钟周期都用黑圆点表示)和对应于这些状态之间的转移支路。

现在考虑图 8.18 所示的卷积码,其中 $k_0=1$, $n_0=2$ 和 $L=3$。很显然,这个码可以用具有 4 个状态的有限状态机表示,这 4 个状态对应于该移位寄存器前两个单元的不同可能内容,即 00,01,10 和 11。现在我们用字母 a,b,c 和 d 分别代表这 4 种状态。为了画出这个码的网格图,必须对每个时钟周期对应于每种状态画出 4 个圆点,然后按照在各状态之间可能发生的各种转移将这些点连起来。对于这个码的网格图见图 8.19。

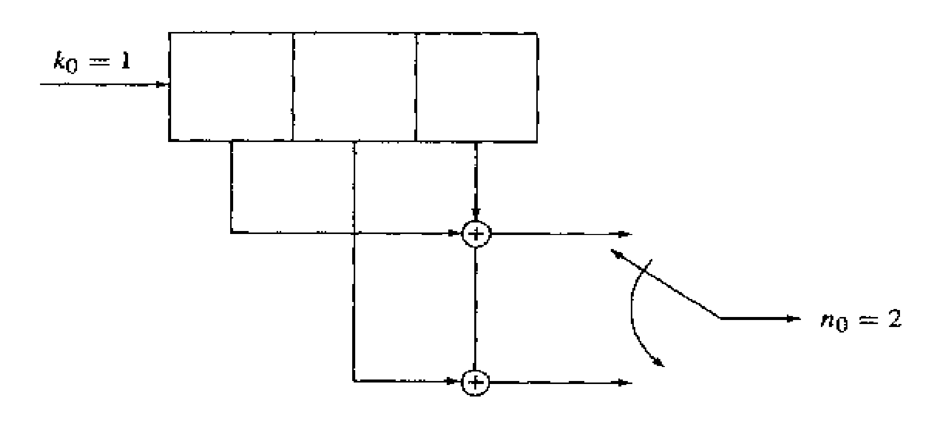


图 8.18　当 $k_0=1$, $n_0=2$ 和 $L=3$ 时的一种卷积编码器

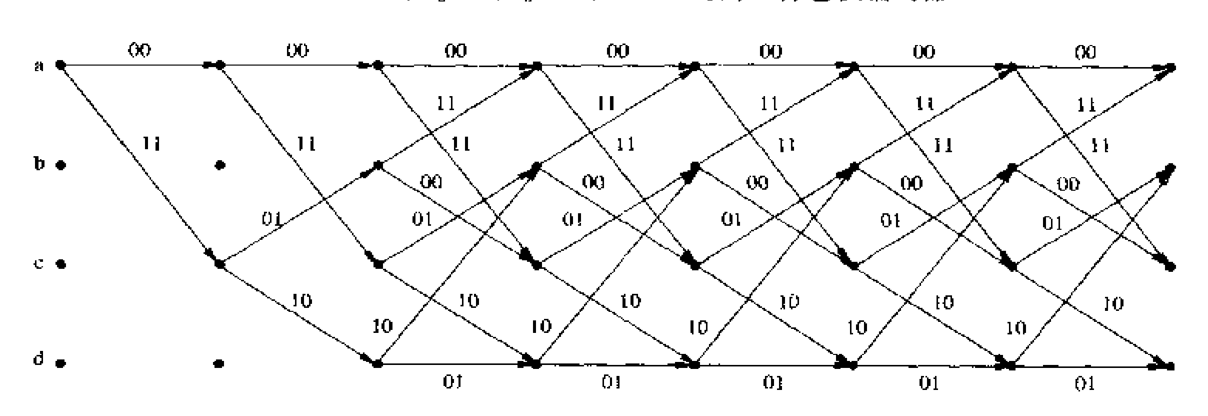


图 8.19　图 8.18 所示的卷积码的网格图

正如在图 8.19 中所看到的,在时间轴上(对应于时钟周期),这 4 种状态是用黑圆点表示的,而状态之间的转移是用连接这些点的支路指出的。在连接两个状态的每条支路上,两个二进制符号指出对应于哪个转移的编码器输出。另外也要注意到,我们总是从全 0 状态(状态 a)出发,沿着对应于给定输入序列的支路通过网格图移动,并回到全 0 状态。因此,一个卷积码的码字就对应于经由相应网格图的路径,它从全 0 状态出发,又回到全 0 状态。

在网格图中的状态数随卷积码的约束长度呈指数增加。例如,对于图 8.17 所示的卷积编码器,状态数是 $2^6=64$,因此这个网格图的结构就会复杂得多。

卷积码的传递函数

对于每个卷积码,传递函数给出了关于通过网格图的各种路径的信息,这些路径从全 0 状态出发并第一次回到这个状态。根据前面所述的编码规则,一个卷积编码器的任何码字都对应于通过网格图从全 0 状态出发并回到这个全 0 状态的一条路径。稍后我们将会知道,一个

卷积码的传递函数在界定这个码的差错概率中起着主要的作用。为了求得卷积码的传递函数,将全0状态分为两种状态,一种代表起始状态,另一种代表第一次回到的全0状态。其余的所有状态都代表中间状态。对应于连接两个状态的每条支路,我们定义一个 $D^{\alpha}N^{\beta}J$ 形式的函数,其中 α 代表在输出比特序列中1的个数,β 是在那条支路的相应输入序列中1的个数。那么,该卷积码的传递函数就是在由全0状态出发到最后的全0状态之间流图的传递函数,并用 $T(D,N,J)$表示。$T(D,N,J)$中的每一项都对应于通过网格图的从全0状态出发并在全0状态结束的一条支路。J 的指数指出跨越这条路径的支路数,D 的指数表明对应于这条路径的码字中1的个数(或等效地说,这个码字与全0码字的 Hamming 距离),而 N 的指数指出在输入信息序列中1的个数。$T(D,N,J)$指出通过这个网格图第一次从全0路径出发并回到全0的所有路径的性质,所以在导出过程中,任何在全0状态的自回路都不考虑。为了得到这个卷积码的传递函数,可以应用在求一个流图传递函数时所用到的全部规则。有关导出一个卷积码的传递函数的详细材料可参阅参考文献[1]

依据导出传递函数的规则,很容易证明图8.18所示的码的传递函数:

$$T(D,N,J)=\frac{D^5NJ^3}{1-DNJ-DNJ^2}$$

当将其展开时可以表示成:

$$T(D,N,J)=D^5NJ^3+D^6N^2J^4+D^6N^2J^5+D^7N^3J^5+\cdots$$

从这个 $T(D,N,J)$的表达式中可以看到,这里有 Hamming 重量为5的一个码字,Hamming 重量为6的两个码字,等等。同时还可以看出,Hamming 重量为5的这个码字对应于 Hamming 重量为1和长度为3的输入序列。在 $T(D,N,J)$的展开式中,D 的最小次方称为卷积码的**自由距离**(free distance),并用 d_{free}表示。在这个例子中,$d_{free}=5$。

卷积码的解码

已经有许多算法可以用于卷积码的解码。Viterbi 算法可能是应用最为广泛的卷积码解码方法。这个算法特别有意思,因为它是一个最大似然解码算法;一旦接收到信道输出,就通过搜索网格图找出最可能产生这个接收序列的路径。如果采用硬判决解码,这个算法就找到与接收序列在最小 Hamming 距离上的那条路径。

在卷积码的硬判决解码中,我们想要选择一条通过网格图的路径,它的码字(记为 $\boldsymbol{c}$)与量化的接收序列 $\boldsymbol{y}$ 有最小的 Hamming 距离。在硬判决解码中,信道是二进制无记忆的(由信道是无记忆的可以得出噪声假设是白色的这一事实)。因为所期望的路径从全0状态出发,并回到全0状态,我们假定这条路径跨越总条数为 m 的支路;又因为每个支路对应于编码器输出的 n_0 个比特,所以在 $\boldsymbol{c}$ 和 $\boldsymbol{y}$ 中的总比特数是 mn_0。现在,分别用 $\boldsymbol{c}_i$ 和 $\boldsymbol{y}_i$ 表示相应于第 i 条支路的比特序列,其中 $1\leqslant i\leqslant m$,并且每个 $\boldsymbol{c}_i$ 和 $\boldsymbol{y}_i$ 的长度都是 n_0。因此,$\boldsymbol{c}$ 和 $\boldsymbol{y}$ 之间的 Hamming 距离是:

$$d(\boldsymbol{c},\boldsymbol{y})=\sum_{i=1}^{m}d(\boldsymbol{c}_i,\boldsymbol{y}_i) \tag{8.3.33}$$

在软判决解码中,除了以下三个方面的不同以外,其余的情况都是类似的:

1. 是用 $\boldsymbol{y}$,而是直接处理向量 $\boldsymbol{r}$,即最佳数字解调器(匹配滤波器型或相关器型)的向量输出。
2. 不用二进制0,1的序列 $\boldsymbol{c}$,而是处理对应的序列 $\boldsymbol{c}'$,有:

$$c'_{ij}=\begin{cases}\sqrt{\mathscr{E}}, & 若\ c_{ij}=1\\ -\sqrt{\mathscr{E}}, & 若\ c_{ij}=0\end{cases}$$

其中，$1\leqslant i\leqslant m$ 和 $1\leqslant j\leqslant n$。

3. 不用 Hamming 距离，而是用欧氏距离。这是基于研究的信道是 AWGN 信道这一事实的一个结果。

根据上面的讨论有：

$$d_{\mathrm{E}}^2(\boldsymbol{c}',\boldsymbol{r})=\sum_{i=1}^{m}d_{\mathrm{E}}^2(\boldsymbol{c}'_i,\boldsymbol{r}_i) \tag{8.3.34}$$

由式(8.3.33)和式(8.3.34)可见，我们必须要解决的这个问题的一般形式是：给定一个向量 $\boldsymbol{a}$，找到一条通过网格图从全 0 状态出发并在全 0 状态终止的一条路径，以使 $\boldsymbol{a}$ 和对应于这条路径的某一序列 $\boldsymbol{b}$ 之间的某个距离测度最小。使这个问题容易求解的重要一点是：在两种令人感兴趣的情况下，$\boldsymbol{a}$ 和 $\boldsymbol{b}$ 之间的距离可写成对应这条路径各单个支路的距离之和。这一点很容易由式(8.3.33)和式(8.3.34)看出。

现在，假设正在处理一个 $k_0=1$ 的卷积码。这就意味着在网格图中仅有两条支路进入每个状态。如果在某一点的最佳路径通过状态 S，就有两条路径把先前的状态 S_1 和 S_2 连接到这个状态（见图 8.20）。

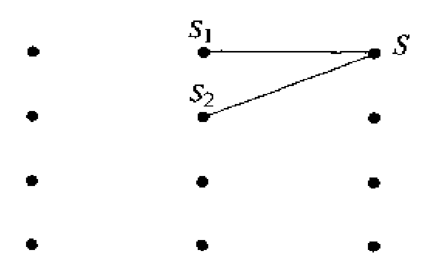


图 8.20　Viterbi 算法的验证

如果我们想知道这两条路径中的哪一条对使总距离最小来说是一个好的候选，就必须将状态 S_1 和 S_2 的总(最小)度量加到把这两个状态连接到状态 S 的这些支路的度量上。很显然，直到状态 S 之前具有最小总度量累加的那条支路是在状态 S 之后要考虑的一条候选支路。这条支路称为在状态 S 的一条**留存支路**(survivor)，而其余的支路不是合适的候选并被删除。现在，当确定了状态 S 的留存支路以后，也就保留了直到这个状态为止的最小度量，就能向前移到下一个状态。这个过程一直继续到网格图末端的全 0 状态为止。对于 $k_0>1$ 的情况，惟一的不同是在每一步都必须从连接到状态 S 的 2^{k_0} 条支路中选出一条留存路径。

上述过程可以综合成下面的算法，即 **Viterbi 算法**：

1. 将接收到的序列分成每段长为 n_0 的 m 组子序列。
2. 对所研究的码画出深度为 m 级的网格图。对该网格图的最后($L-1$)级，仅画出对应于全 0 输入序列的路径(这样做是因为知道输入序列已经给补了 $k_0(L-1)$个零)。
3. 置 $l=1$，并置初始全 0 状态的度量等于 0。
4. 对网格图中把第 l 级状态连接到第($l+1$)级状态的所有支路，求出该接收序列中第 l 个子序列的距离。

5. 将这些距离加到第 l 级的各状态的度量上,得到对第($l+1$)级状态的度量候选,对于第($l+1$)级的每个状态,有 2^{k_0} 个候选度量,其中每个都对应终止在那个状态的一条支路。
6. 对在第($l+1$)级的每个状态,挑选出最小的候选度量,并将对应于这个最小值的支路标为**留存支路**,同时指定这个候选度量的最小值作为第($l+1$)级状态的度量。
7. 若 $l=m$,转到下一步,否则将 l 加 1 并转到第 4 步。
8. 在第($m+1$)级以全 0 状态开始,沿着留存支路通过网格图回到初始全 0 状态,这条路径就是最佳路径,并且对应于这条路径的输入比特序列是最大似然解码信息序列。为了得到有关这个输入比特序列的最好推测,将最后 $k_0(L-1)$个 0 从该序列中除掉。

从这个算法中可以看到,为一个长信息序列解码时,解码延时和所需的存储量都是无法接受的。直到整个序列全都被接收,才能开始解码(在卷积码的情况下,时间可能会很长),而且还不得不将总的留存路径存储起来。实际上,不会引起这些问题的次优解倒是我们所期望的。一种称为**路径存储截断**(path memory truncation)的办法是:对于每一级编码器,在网格图中仅往回搜索 δ 级,而不回到网格图的出发点。使用这种办法,在第($\delta+1$)级,解码器对相应于网格图 1 级的输入比特(第一个 k_0 比特)做出判决,并且未来的接收比特不改变这个判决。这意味着解码延时是 $k_0\delta$ 比特,需要保留的路径只相应于最后 δ 级的路径。计算机仿真已经表明,如果 $\delta \geqslant 5L$,由于路径存储截断而造成的性能下降可以忽略不计。

解说题

解说题 8.14　[Viterbi 解码]

假定在硬判决解码下,量化的接收序列是:

$$\mathbf{y}=01101111010001$$

卷积码由图 8.18 给出。求最大似然信息序列和差错数。

题　解

这个码是 $L=3$ 的(2,1)码。接收序列 $\mathbf{y}$ 的长度是 14,这就意味着 $m=7$,需要画 7 级的网格图。同时我们也注意到,因为输入信息序列补了 $k_0(L-1)=2$ 个 0,因此在网格图的最后两级仅需画出对应于全 0 输入的支路。这也意味着真正的输入序列的长度是 5,在补了两个 0 后就增加到 7。这种情况的网格图见图 8.21。

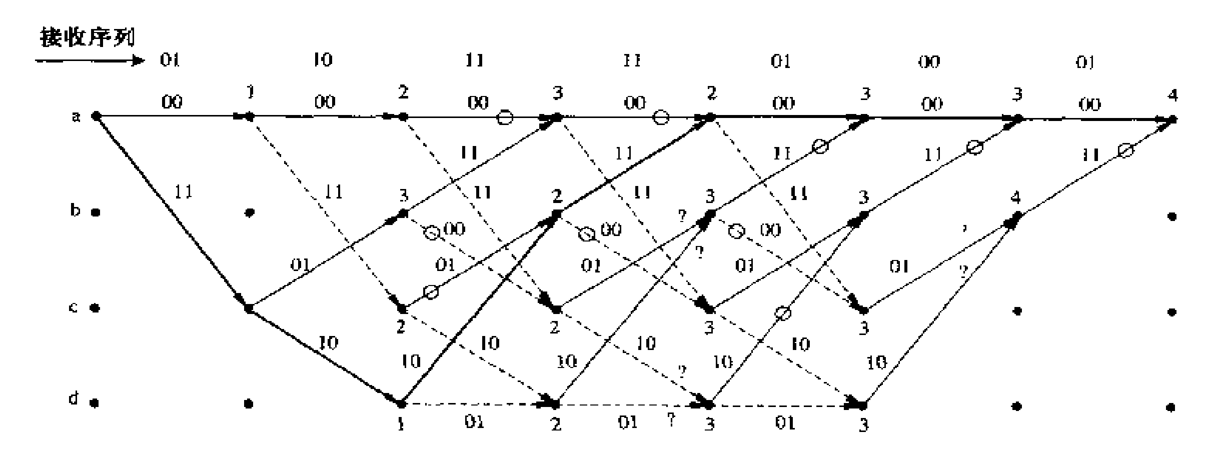


图 8.21　对序列(01101111010001)进行 Viterbi 解码的网格图

图 8.21 中也给出了划分后的接收序列 $\mathbf{y}$。注意，在画这个网格图时，最后两级已经考虑到对编码器仅有 0 输入(注意，在最后两级不存在对应于 1 输入的虚线)。现在，初始全 0 状态的度量置为 0，并计算出下一级的度量。在这一步，只有一条支路进入每个状态，因此也不存在任何比较，将这些度量(它们就是接收序列部分和网格支路之间的 Hamming 距离)加到先前状态的度量上。在下一级仍不存在任何比较。在第 4 级，第一次有进入每个状态的两条支路。这意味着在这里必须做出比较，并挑选出留存支路。从进入每个状态的两条支路中，对应于最小总累加度量的一条支路保留下来作为留存支路，而另一条支路就被剔除掉了(在该网格图上用小圆圈标出)。如果在任何一级，两条路径得出相同的度量，那么它们中的每一条都能作为一条留存路径。这种情况在这个网格图中已经用问号"?"表示出来了。这个过程一直持续到网格图最后的全部 0 状态，然后从那个状态出发，沿着留存路径移动到最初的全 0 状态。这条路径的输入比特序列是 1100000，其中最后两个 0 不是信息比特，而是被加上的，以使编码器回到全 0 状态。因此，这个信息序列是 11000。对应于挑选出的路径的码字是 11101011000000，它与接收序列的 Hamming 距离是 4。通过这个网格图的所有其他路径对应的码字与接收序列都有更大的 Hamming 距离。

对于软判决解码，用平方欧氏距离代替 Hamming 距离，也有一个类似的过程。

下面给出的 MATLAB 函数 viterbi.m 使用 Viterbi 算法对信道输出解码。这个算法既能用于卷积码的软判决解码，也能用于硬判决解码。单独分开的文件 metric.m 定义了在解码过程中所用的度量标准。对于硬判决解码，这个度量标准是 Hamming 距离；对于软判决解码，这个标准是欧氏距离。对于信道输出已量化的情况，这个度量标准通常是对数似然率的负值，即 $-\log p$(信道输出|信道输入)。下面也给出了函数 viterbi.m 调用的几个较短的 m 文件。

m 文件

```
function [decoder_output,survivor_state,cumulated_metric]=viterbi(G,k,channel_output)
%VITERBI      The Viterbi decoder for convolutional codes
%             [decoder_output,survivor_state,cumulated_metric]=viterbi(G,k,channel_output)
%             G is a n x Lk matrix each row of which
%             determines the connections from the shift register to the
%             n-th output of the code. k/n is the rate of the code.
%             survivor_state is a matrix showing the optimal path through
%             the trellis. The metric is given in a separate function metric(x,y)
%             and can be specified to accommodate hard and soft decision.
%             This algorithm minimizes the metric rather than maximizing
%             the likelihood.

n=size(G,1);
%  check the sizes
if rem(size(G,2),k) ~=0
  error('Size of G and k do not agree')
end
if rem(size(channel_output,2),n) ~=0
  error('channel output not of the right size')
end
L=size(G,2)/k;
number_of_states=2^((L-1)*k);
%  Generate state transition matrix, output matrix, and input matrix.
for j=0:number_of_states -1
  for l=0:2^k -1
```

```
    [next_state,memory_contents]=nxt_stat(j,l,L,k);
    input(j+1,next_state+1)=l;
    branch_output=rem(memory_contents*G',2);
    nextstate(j+1,l+1)=next_state;
    output(j+1,l+1)=bin2deci(branch_output);
  end
end
state_metric=zeros(number_of_states,2);
depth_of_trellis=length(channel_output)/n;
channel_output_matrix=reshape(channel_output,n,depth_of_trellis);
survivor_state=zeros(number_of_states,depth_of_trellis+1);
%  Start decoding of non-tail channel outputs.
for i=1:depth_of_trellis-L+1
  flag=zeros(1,number_of_states);
  if i <= L
    step=2^((L-i)*k);
  else
    step=1;
  end
  for j=0:step:number_of_states-1
    for l=0:2^k-1
      branch_metric=0;
      binary_output=deci2bin(output(j+1,l+1),n);
      for ll=1:n
        branch_metric=branch_metric+metric(channel_output_matrix(ll,i),binary_output(ll));
      end
      if((state_metric(nextstate(j+1,l+1)+1,2) > state_metric(j+1,1)...
        +branch_metric) | flag(nextstate(j+1,l+1)+1)==0)
        state_metric(nextstate(j+1,l+1)+1,2) = state_metric(j+1,1)+branch_metric;
        survivor_state(nextstate(j+1,l+1)+1,i+1)=j;
        flag(nextstate(j+1,l+1)+1)=1;
      end
    end
  end
  state_metric=state_metric(:,2:-1:1);
end
%  Start decoding of the tail channel outputs.
for i=depth_of_trellis-L+2:depth_of_trellis
  flag=zeros(1,number_of_states);
  last_stop=number_of_states/(2^((i-depth_of_trellis+L-2)*k));
  for j=0:last_stop-1
      branch_metric=0;
      binary_output=deci2bin(output(j+1,1),n);
      for ll=1:n
        branch_metric=branch_metric+metric(channel_output_matrix(ll,i),binary_output(ll));
      end
      if((state_metric(nextstate(j+1,1)+1,2) > state_metric(j+1,1)...
        +branch_metric) | flag(nextstate(j+1,1)+1)==0)
        state_metric(nextstate(j+1,1)+1,2) = state_metric(j+1,1)+branch_metric;
        survivor_state(nextstate(j+1,1)+1,i+1)=j;
        flag(nextstate(j+1,1)+1)=1;
      end
  end
  state_metric=state_metric(:,2:-1:1);
end
%  Generate the decoder output from the optimal path.
state_sequence=zeros(1,depth_of_trellis+1);
state_sequence(1,depth_of_trellis)=survivor_state(1,depth_of_trellis+1);
for i=1:depth_of_trellis
```

```
  state_sequence(1,depth_of_trellis-i+1)=survivor_state((state_sequence(1,depth_of_trellis+2-i)...
  +1),depth_of_trellis-i+2);
end
decodeder_output_matrix=zeros(k,depth_of_trellis-L+1);
for i=1:depth_of_trellis-L+1
  dec_output_deci=input(state_sequence(1,i)+1,state_sequence(1,i+1)+1);
  dec_output_bin=deci2bin(dec_output_deci,k);
  decoder_output_matrix(:,i)=dec_output_bin(k:-1:1)';
end
decoder_output=reshape(decoder_output_matrix,1,k*(depth_of_trellis-L+1));
cumulated_metric=state_metric(1,1);
```

m 文件

```
function distance=metric(x,y)
if x==y
  distance=0;
else
  distance=1;
end
```

m 文件

```
function [next_state,memory_contents]=nxt_stat(current_state,input,L,k)
binary_state=deci2bin(current_state,k*(L-1));
binary_input=deci2bin(input,k);
next_state_binary=[binary_input,binary_state(1:(L-2)*k)];
next_state=bin2deci(next_state_binary);
memory_contents=[binary_input,binary_state];
```

m 文件

```
function y=bin2deci(x)
l=length(x);
y=(l-1:-1:0);
y=2.^y;
y=x*y';
```

m 文件

```
function y=deci2bin(x,l)
y = zeros(1,l);
i = 1;
while x>=0 & i<=l
        y(i)=rem(x,2);
        x=(x-y(i))/2;
        i=i+1;
end
y=y(l:-1:1);
```

解说题

解说题 8.15

用 MATLAB 函数 viterbi.m 重做解说题 8.14。

题 解

根据下列输入,利用 m 文件 viterbi.m:

$$G=\begin{bmatrix}1 & 0 & 1\\ 1 & 1 & 1\end{bmatrix}$$

$$k=1$$

$$\text{channel_output}=[0\quad 1\quad 1\quad 0\quad 1\quad 1\quad 1\quad 1\quad 0\quad 1\quad 0\quad 0\quad 0\quad 1]$$

即可得到 decoder_output = [1 1 0 0 0],累加的度量是 4。

卷积码差错概率的界

由于卷积码序列是一个很长的序列,所以寻找卷积码差错性能的上界所采用的方法不同于在分组码中所用的方法;因为这些码的自由距离通常都很小,所以有些差错最终会发生。差错数是一个随机变量,它既与信道特性(软判决解码时的信噪比和硬判决解码时的交叉概率)有关,又与序列的长度有关。序列愈长,产生差错的概率愈大,因此将比特差错数对输入序列长度归一化才会有意义。通常,为比较卷积码性能而采用的一种度量是每输入比特接收到的差错比特的期望数。为了找到对每输入比特的差错比特平均数的界,首先求出对每个长为 k 的输入序列的差错比特平均数的界。为了确定这个界,假设传输的是全 0 序列[①],并且直到解码的第 l 级为止,都没有任何差错。现在,k 个信息比特进入到编码器并在网格图中朝下一级移动。我们关心的是找到一个由这个长为 k 的输入组能产生的期望差错数的界。因为已假设直到 l 级为止都没有任何误差,所以到这一级为止,通过网格图的全 0 路径有最小的度量。现在当我们朝下一级,即第 $(l+1)$ 级移动时,就可能有另一条通过网格图的路径,其度量比这个全 0 路径还要小,因此会引起误差。如果发生这种情况,就必须有通过网格图的某一条路径,该路径第 1 次在第 $(l+1)$ 级与全 0 路径合并,并有一个小于全 0 路径的度量。这样的一个事件称为**首次差错事件**(first error event),相应的概率称为**首次差错事件概率**(first error event probability)。这种情况如图 8.22 所示。

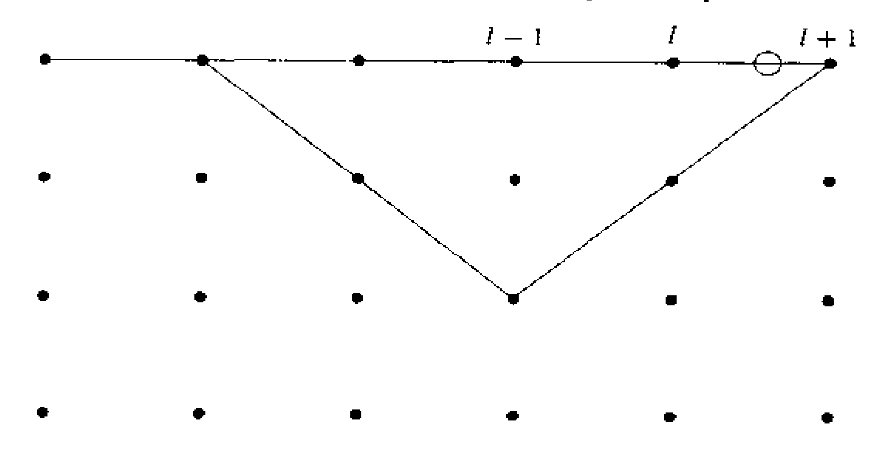


图 8.22 对应于首次差错事件的路径

① 由于卷积码的线性特性,可以做这个假设而不失一般性。

第一步是要界定这个首次差错事件的概率。令 $P_2(d)$代表在第$(l+1)$级的留存路径的概率,它是通过网格图的与全 0 路径的 Hamming 距离为 d 的一条路径。因为 d 比 d_{free}大,所以可将首次差错事件概率界定为:

$$P_e \leqslant \sum_{d=d_{\text{free}}}^{\infty} a_d P_2(d)$$

上式右边已经包括了在第$(l+1)$级与全 0 路径合并的通过网格图的所有路径。$P_2(d)$代表与全 0 路径有 Hamming 距离 d 的一条路径的差错概率,a_d 代表与全 0 路径有 Hamming 距离 d 的路径数目。$P_2(d)$的值取决于是采用软判决解码,还是采用硬判决解码。

对于软判决解码,如果采用的是反极性信号(二进制 PSK),就有:

$$\begin{aligned}P_2(d) &= Q\left(\frac{d^E}{\sqrt{2N_0}}\right)\\ &= Q\left(\sqrt{\frac{2\mathscr{E}\ d}{N_0}}\right)\\ &= Q\left(\sqrt{2R_c d\frac{\mathscr{E}_b}{N_0}}\right)\end{aligned}$$

因此,

$$P_e \leqslant \sum_{d=d_{\text{free}}}^{\infty} a_d Q\left(\sqrt{2R_c d\frac{\mathscr{E}_b}{N_0}}\right)$$

利用在 Q 函数上已知的上界

$$Q(x) \leqslant \frac{1}{2}e^{-x^2/2}$$

可得:

$$Q\left(\sqrt{2R_c d\frac{\mathscr{E}_b}{N_0}}\right) \leqslant \frac{1}{2}e^{-R_c d\mathscr{E}_b/N_0}$$

现在,注意到:

$$e^{-R_c d\mathscr{E}_b/N_0} = D^d\Big|_{D=e^{-R_c\mathscr{E}_b/N_0}}$$

最后得到:

$$P_e \leqslant \frac{1}{2}\sum_{d=d_{\text{free}}}^{\infty} a_d D^d\Big|_{D=e^{-R_c\mathscr{E}_b/N_0}} = \frac{1}{2}T_1(D)\Big|_{D=e^{-R_c\mathscr{E}_b/N_0}}$$

其中

$$T_1(D) = T(D,N,J)\big|_{N=J=1}$$

这是一个在首次差错事件概率上的界。为了求得对 k 输入比特的差错比特平均数的界 $\bar{P}_b(k)$,我们注意到通过网格图的每条路径都会产生某输入比特数被错误地解码。在 $T(D,N,J)$展开式的一般表示 $D^d N^{f(d)} J^{g(d)}$① 中,总共有 $f(d)$个非零输入比特。这就意味着,把挑选出的每条路径的概率乘以被选路径所产生的总输入差错数,就可以得到差错输入比特的平

① 这里在符号上稍许有些粗糙。严格地说,N 的方次不是 d 的函数,但是用 $f(d)$来表示。不过,对于最后结果并没有任何影响。

均数。所以,在软判决情况下,差错比特的平均数被界定为:

$$\begin{aligned}P_b(k) &\leqslant \sum_{d=d_{free}}^{\infty} a_d f(d) P_2(d)\\ &= \sum_{d=d_{free}}^{\infty} a_d f(d) Q\left(\sqrt{2R_c d\frac{\mathscr{E}_b}{N_0}}\right)\\ &\leqslant \frac{1}{2}\sum_{d=d_{free}}^{\infty} a_d f(d) e^{-R_c d\mathscr{E}_b/N_0}\end{aligned} \tag{8.3.35}$$

如果定义:

$$T_2(D,N) = T(D,N,J)\big|_{J=1} = \sum_{d=d_{free}}^{\infty} a_d D^d N^{f(d)}$$

就有:

$$\left.\frac{\partial T_2(D,N)}{\partial N}\right|_{N=1} = \sum_{d=d_{free}}^{\infty} a_d f(d) D^d \tag{8.3.36}$$

因此,利用式(8.3.35)和式(8.3.36),可得:

$$\bar{P}_b(k) \leqslant \frac{1}{2}\left.\frac{\partial T_2(D,N)}{\partial N}\right|_{N=1,D=e^{-R_c\mathscr{E}_b/N_0}}$$

为了求得对每输入比特的差错比特平均数,必须将这个界除以 k,由此得到的最后结果是:

$$\bar{P}_b = \frac{1}{2k}\left.\frac{\partial T_2(D,N)}{\partial N}\right|_{N=1,D=e^{-R_c\mathscr{E}_b/N_0}}$$

对于硬判决解码,基本过程与上面的推导相仿,惟一的不同是在 $P_2(d)$上的界。可以证明(见参考文献[1]),$P_2(d)$由下式界定为:

$$P_2(d) \leqslant [4p(1-p)]^{d/2}$$

由这个结果可以直接证明,在硬判决解码下,差错概率的上界是:

$$\bar{P}_b \leqslant \frac{1}{k}\left.\frac{\partial T_2(D,N)}{\partial N}\right|_{N=1,D=\sqrt{4p(1-p)}}$$

对卷积码将硬判决解码和软判决解码进行比较可以得出,和在线性分组码的情况下相同,在加性高斯白噪声信道下,软判决解码在性能上优于硬判决解码大致 2~3 dB 的范围。

8.4 习题

8.1 写一个 MATLAB 程序,画出具有交叉概率 p 的二进制对称信道的容量作为 $p(0 \leqslant p \leqslant 1)$ 的函数的图。当 p 为何值时容量最小?最小值是多少?

8.2 一个二进制非对称信道用条件概率 $p(0|1)=0.2$ 和 $p(1|0)=0.4$ 表征,画出该信道输入和输出之间的互信息 $I(X;Y)$作为 $p=P(X=1)$的函数的图。当 p 为何值时互信息最大?最大值是多少?

8.3 有一个 Z 信道是一个二进制输入、二进制输出信道,其输入和输出字符为 $\mathscr{X}=\mathscr{Y}=\{0,1\}$,并由 $p(0|1)=\varepsilon$ 和 $p(1|0)=0$ 表征。对 $\varepsilon=0,0.1,0.2,0.3,0.4,0.5,0.7,0.9,1$,画出 $I(X;Y)$作为 $p=P(X=1)$的函数的图。求每种情况下信道的容量。

8.4 一个二进制输入、三元输出信道由输入和输出字符集 $\mathscr{X}=\{0,1\}$和 $\mathscr{Y}=\{0,1,2\}$,转移概率

为 $p(0|0)=0.05, p(1|0)=0.2, p(0|1)=0.1$ 和 $p(1|1)=0.1$ 表征。画出 $I(X;Y)$ 作为 $p=P(X=1)$ 的函数的图，并求信道容量。

8.5　一个三元输入、二进制输出信道由输入和输出字符集 $\mathcal{X}=\{0,1,2\}$ 和 $\mathcal{Y}=\{0,1\}$，转移概率 $p(0|0)=0.05, p(1|1)=0.2$ 和 $p(0|2)=0.1$ 表征。画出 $I(X;Y)$ 作为 $p_1=P(X=1)$ 和 $p_2=P(X=2)$ 的函数的图，并求信道容量。

8.6　画出使用二进制正交信号的二进制对称信道的容量作为 $\mathcal{E}_b/N_0$ 的函数的图。

8.7　假定两个传输信号是等能量且为正交的，重做解说题 8.3。所得结果与在解说题 8.3 中所得结果有什么不同？

8.8　当使用正交信号时，比较在硬判决和软判决下的容量图，并将这些结果与在使用反极性信号时所得结果进行比较。

8.9　画出使用正交信号时二进制对称信道的容量作为 $\mathcal{E}_b/N_0$ 的函数的图，在求解时一次用相干检测的假设，一次用非相干检测的假设。将这两个图展示在同一个图上，并比较这两个结果。

8.10　写一个 MATLAB 程序，产生对任何已知 m 的规则形式的 Hamming 码生成矩阵。

8.11　利用正交信号，分别用相干和非相干检测重做解说题 8.10，并在同一个图上画出这两个结果。

8.12　利 Monte Carlo 仿真画出解说题 8.10 中差错概率与 γ_b 的对比图。

8.13　不用正交信号而用反极性信号重做解说题 8.12。在软判决解码下，比较正交信号的相干和非相干解调的性能。

8.14　利用 Monte Carlo 仿真，画出解说题 8.12 中差错概率与 γ_b 的对比图。

8.15　当输入序列是

1 1 0 0 1 0 1 0 1 0 1 0 0 1 0 1 1 1 1 0 1 0 1 1 1 1 1 0 1 0

时，用 MATLAB 求图 8.18 所示的卷积编码器的输出。

8.16　一个卷积码由生成矩阵

$$G=\begin{bmatrix}1 & 0 & 1 & 1\\ 0 & 1 & 1 & 0\\ 1 & 1 & 0 & 1\\ 1 & 1 & 1 & 1\end{bmatrix}$$

描述

(a) 若 $k=1$，当输入序列是

1 1 0 0 1 0 1 0 1 0 1 0 0 1 0 1 1 1 1 0 1 0 1 1 1 1 1 0 1 0

时，求该编码器的输出。

(b) 用 $k=2$，重做(a)。

8.17　在习题 8.15 中，求得该编码器的输出之后，改变接收序列的前 6 个比特，并用 Viterbi 解码解出这一结果。将这个解码器输出与传输序列进行比较。发生了多少个差错？一次改变接收序列的最后 6 个比特，一次改变接收序列的前 3 个比特，重做这个习题，并比较这些结果。在全部过程中都用 Hamming 度量标准。

8.18　产生一个长为 1000 的等概率二进制序列，并将该序列按图 8.18 所示的卷积码编码。产

生 4 个随机二进制差错序列,每个长为 2000,其 1 的概率分别等于 0.01,0.05,0.1 和 0.2。将这些差错序列中的每个加(模 2)到这个已编码的序列上,并用 Viterbi 算法解码出结果。在每种情况下,将这个解码序列与编码器输入进行比较,并求出误码率。

8.19 利用图 8.18 所示的码,通过 Monte Carlo 仿真画出在某卷积编码器中误码率与 γ_b 的对比图。假设采用二进制反极性调制方法,一次用硬判决解码,一次用软判决解码。令 γ_b 位于 3~9 dB 的区间内,并适当地选取信息序列长度。将得到的结果与理论上的界进行比较。

8.20 在一条具有二元输入和三元输出的信道上,用图 8.18 所示的编码器传输信息。输出用 0,1 和 2 表示。这就属于一个高斯信道的输出被量化到 3 个电平的情况。信道的条件概率由 $p(0|0)=p(1|1)=0.9$, $p(2|0)=p(2|1)=0.09$ 给出。用 Viterbi 算法对下面的接收序列解码:

0 2 0 2 0 1 1 0 2 0 2 1 1 2 0 0 2 2 2 0 1 1 0 1 0 1 0 2 0 2 2 0 1 1 1 1 1 2

第 9 章　扩频通信系统

9.1　概述

数字通信中的扩频信号最初是为军事通信的需要而发展和应用起来的，其中包括：(1) 对抗敌方干扰；(2) 用在低功率下传输以隐藏信号，使得无意的听众在噪声中不易检测出信号的存在；(3) 有可能经由同一信道进行多用户通信。然而，今天扩频信号正在为各种商业应用提供可靠的通信，其中包括车载移动通信和办公室之间的无线通信。

一个扩频数字通信系统的基本组成如图 9.1 所示。可以看到，信道编码器和解码器、调制器和解调器都是常规数字通信系统的基本环节。除了这些以外，扩频系统中还使用了两个完全一样的伪随机序列发生器，其中一个在发送端与调制器相接，另一个在接收端与解调器相接。这两个发生器产生伪随机或伪噪声(PN)的二值序列，用于在调制器中将传输信号在频率上进行扩频和在解调器上将接收信号进行解扩。

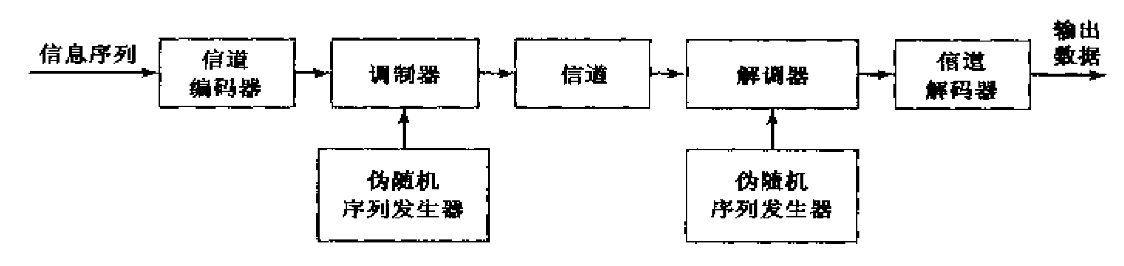


图 9.1　扩频数字通信系统模型

为正确地对接收到的扩频信号进行解扩，要求在接收机产生的 PN 序列与包含在接收信号中的 PN 序列同步。在实际的系统中，同步在信息传输之前已经建立，这是通过传输一个专门设计的固定的 PN 比特样本实现的，以使接收机能在干扰存在的情况下以高的概率检测到它。在这个 PN 序列发生器的时间同步建立之后，再开始信息传输。在数据模式上，通常该通信系统跟踪正在到来的接收信号的定时，并将这个 PN 序列发生器保持在同步状态。

这一章要讨论两种基本的用于数字通信中的扩频信号类型，即直接序列(DS)扩频信号和跳频(FH)扩频信号。

我们要考虑与扩频信号相关的两种数字调制型式：PSK 和 FSK。PSK 调制一般在 DS 扩频中应用，并且最适合用在这样的场合：发送信号和接收信号之间的相位相干性可以在横跨几个符号(或比特)区间的一段时间内保持住。另一方面，FSK 调制一般用在 FH 扩频中，并且最适合用在这样的场合：由于通信信道传输特性的时变性不能保持，不能保持载波的相位相干性。

9.2　直接序列扩频系统

现在来考虑采用二进制 PSK 的一个二进制信息序列的传输问题。信息速率是 R bps，比

特区间是 $T_b = 1/R$ s。可用的信道带宽是 B_c(Hz)，$B_c >> R$。在调制器中，按照 PN 发生器的变化形式，以每秒 W 次的速率，伪随机地将载波相位移相，将信息信号的带宽扩展到 $W = B_C$ (Hz)。这样，所得到的已调信号称为**直接序列**(DS)**扩频信号**。

载有信息的基带信号用 $v(t)$ 代表，可表示为：

$$v(t) = \sum_{n=-\infty}^{\infty} a_n g_T(t - nT_b) \tag{9.2.1}$$

其中，$\{a_n = \pm 1, -\infty < n < \infty\}$，$g_T(t)$ 是持续期为 T_b 的矩阵脉冲。这个信号乘以从 PN 序列发生器来的信号，该信号可以表示为：

$$c(t) = \sum_{n=-\infty}^{\infty} c_n p(t - nT_c) \tag{9.2.2}$$

其中，$\{c_n\}$ 代表该二进制 PN 的值为 ±1 的码序列，$p(t)$ 是持续期为 T_c 的矩形脉冲，如图 9.2 所示。这个相乘运算用来将载有信息的信号带宽(大约为 R Hz)扩展到 PN 发生器信号 $c(t)$ 所占的更宽的带宽(这个带宽大约为 $1/T_c$)上。频谱的扩展如图 9.3 所示，图中用简单的矩形谱给出了两个谱的卷积。较窄的谱对应于载有信息的信号，较宽的谱对应于来自 PN 发生器的信号。

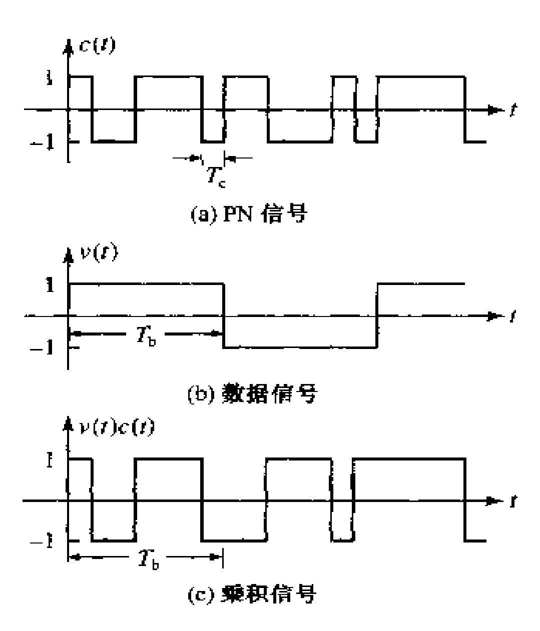


图 9.2　DS 扩频信号的产生

该乘积信号 $v(t)c(t)$(已在图 9.2 中给出)用来对载波 $A_c\cos(2\pi f_c t)$ 进行幅度调制，产生的 *DSB-SC* 信号为：

$$u(t) = A_c v(t) c(t) \cos(2\pi f_c t) \tag{9.2.3}$$

因为对任意 t，$v(t)c(t) = \pm 1$，所以这个载波调制传输信号也可以表示为：

$$u(t) = A_c \cos[2\pi f_c t + \theta(t)] \tag{9.2.4}$$

其中，当 $v(t)c(t) = 1$ 时，$\theta(t) = 0$；当 $v(t)c(t) = -1$ 时，$\theta(t) = \pi$。因此，这个传输信号是一个二进制 PSK 信号，它的相位在 $1/T_c$ 速率下变化。

矩形脉冲 $p(t)$ 通常称为**码片**(chip)，它的时间持续期 T_c 称为**码片间隔**(chip interval)，其

倒数 $1/T_c$ 称为**码片速率**(chip rate),并(近似地)相应于传输信号的带宽 W。在实际的扩频系统中,比特区间 T_b 对码片间隔 T_c 的比通常都选为某个整数。我们记这个比值为:

$$L_c = \frac{T_b}{T_c} \tag{9.2.5}$$

所以 L_c 就是每信息比特中 PN 码序列的码片数。另一种解释是,L_c 代表了在比特区间 T_b 期间,在传输信号中可能有的 180°相位转移的数目。

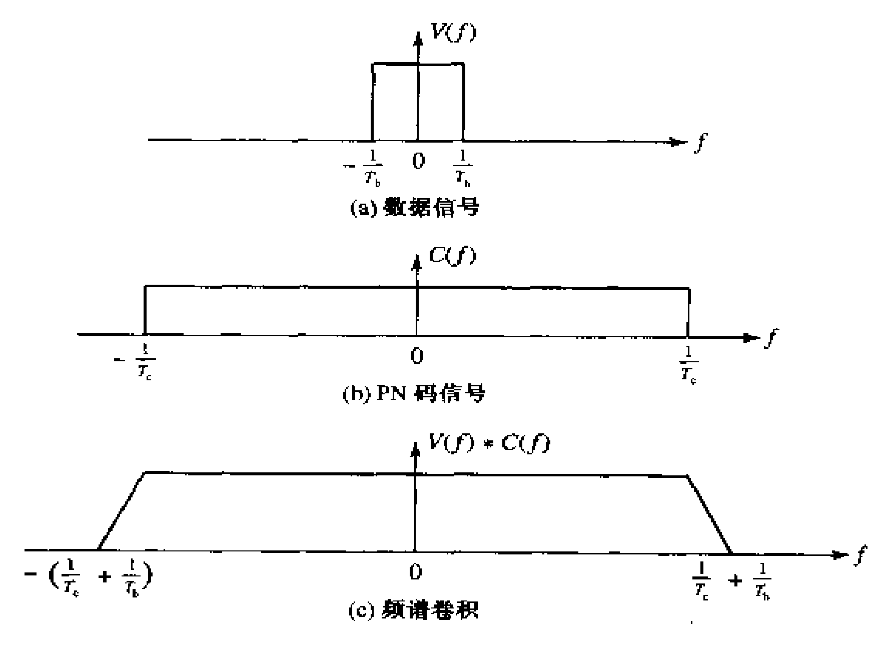


图 9.3　频谱的扩展

9.2.1　信号解调

信号解调按图 9.4 完成。接收信号首先乘以在接收机的 PN 码序列发生器中产生的波形 $c(t)$ 的副本,它是与接收信号中的 PN 码同步的。这一运算称为(频谱)**解扩**,因为在接收端乘以 $c(t)$ 的效果就是将在发送端的扩频运算解开。据此有:

$$A_c v(t) c^2(t) \cos(2\pi f_c t) = A_c v(t) \cos(2\pi f_c t) \tag{9.2.6}$$

上式中因为有对所有的 t,$c^2(t)=1$,所得信号 $A_c v(t)\cos(2\pi f_c t)$ 占有的带宽为 R(Hz)(近似),这就是载有信息信号的带宽。因此,对于解扩信号而言,这个解调器就是常规的互相关器或匹配滤波器,这些在第 5 章和第 7 章中都讨论过了。因为该解调器具有和解扩信号相同的带宽,所以在解调器上使信号受到污损的仅是那些位于接收信号的信息带宽以内的加性噪声。

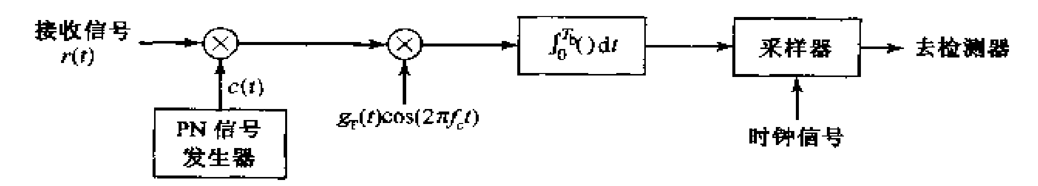


图 9.4　DS 扩频信号的解调

窄带干扰对解扩的影响

研究某一干扰信号对所期望的载有信息信号的解调的影响是很意义的。设想接收信号是：

$$r(t) = A_c v(t) c(t) \cos(2\pi f_c t) + i(t) \tag{9.2.7}$$

其中，$i(t)$代表干扰，在接收端的解扩处理产生：

$$r(t) c(t) = A_c v(t) \cos(2\pi f_c t) + i(t) c(t) \tag{9.2.8}$$

将干扰 $i(t)$与 $c(t)$相乘的效果就是将 $i(t)$的带宽扩频到 W Hz。

作为一个例子－考虑一个如下式的正弦干扰信号：

$$i(t) = A_J \cos(2\pi f_J t) \tag{9.2.9}$$

其中，f_J 是位于传输信号带宽内的某一频率。它与 $c(t)$相乘就会产生一个功率谱密度 $J_0 = P_J / W$ 的宽带干扰，这里 $P_J = A_J^2/2$ 是该干扰的平均功率。因为所期望的信号是用带宽为 R 的匹配滤波器(或相关器)解调的，所以在解调器输出端干扰的总功率是：

$$J_0 R = \frac{P_J R}{W} = \frac{P_J}{W/R} = \frac{P_J}{T_b/T_c} = \frac{P_J}{L_c} \tag{9.2.10}$$

因此，在干扰信号中的功率就降低了，等于带宽扩展因子 W/R 的倍数。这个因子 $W/R = T_b/T_c = L_c$ 称为扩频系统的**处理增益**(processing gain)。干扰功率的降低是在有干扰的信道上采用扩频信号传输数字信息的根本原因。

总之，在发送端用 PN 码序列将载有信息的信号扩频到某个较宽的带宽上，然后在信道上进行传输。当这个接收信号乘以一个同步了的 PN 码信号的副本时，所期望的信号被解扩回到窄带带宽上，而任何干扰信号都被扩频到一个较宽的带宽内。在干扰功率上的净效果就是降低 W/R 倍，这就是扩频系统的处理增益。

假定这个 PN 码序列$\{c_n\}$已知是仅对这个目标中的接收机而言的。任何不具有该 PN 码序列知识的其他接收机不可能解出这个信号。这样，利用一个 PN 码序列就提供了某种程度的保密性(或安全性)，而使用常规的调制是不可能达到这个目的的。为了获得安全性和抗干扰性能，所付出的主要代价是在信道带宽利用上的降低和在通信系统复杂性上的增加。

9.2.2　差错概率

在 AWGN 信道中，采用二进制 PSK 的 DS 扩频系统的差错概率与一般的(未扩频)二进制 PSK 的差错概率是相同的，即

$$P_b = Q\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right) \tag{9.2.11}$$

另外，如果干扰是如式(9.2.9)给出的正弦型的，其功率为 P_J，那么差错概率(近似)为：

$$P_b = Q\left(\sqrt{\frac{2\mathcal{E}_b}{P_J/W}}\right) = Q\left(\sqrt{\frac{2\mathcal{E}_b}{J_0}}\right) \tag{9.2.12}$$

于是干扰功率降低了扩频信号带宽 W 倍的因子。在这种情况下，没有顾及 AWGN(假设可以忽略)，即 $N_0 \ll P_J/W$。如果考虑信道中的 AWGN，则差错概率可以表示为：

$$P_b = Q\left(\sqrt{\frac{2\mathcal{E}_b}{N_0 + P_J/W}}\right)$$

$$= Q\left(\sqrt{\frac{2\mathscr{E}_b}{N_0 + J_0}}\right) \tag{9.2.13}$$

干扰余裕度(Jamming Margin)

可将$\mathscr{E}_b/J_0$表示为：

$$\frac{\mathscr{E}_b}{J_0} = \frac{P_S T_b}{P_J/W} = \frac{P_S/R}{P_J/W} = \frac{W/R}{P_J/P_S} \tag{9.2.14}$$

现在假定给出某一要求的$\mathscr{E}_b/J_0$，以实现某个期望的性能，那么利用对数标尺可将式(9.2.14)表示为：

$$10\log\frac{P_J}{P_S} = 10\log\frac{W}{R} - 10\log\left(\frac{\mathscr{E}_b}{J_0}\right)$$

$$\left(\frac{P_J}{P_S}\right)_{dB} = \left(\frac{W}{R}\right)_{dB} - \left(\frac{\mathscr{E}_b}{J_0}\right)_{dB} \tag{9.2.15}$$

比值$(P_J/P_S)_{dB}$称为**干扰余裕度**。这就是一个干扰台能有的不破坏通信系统的相对功率得益。

解说题

解说题 9.1　[处理增益和干扰余裕度]

假设需要$\mathscr{E}_b/J_0 = 10$ dB 以实现二制 PSK 的可靠通信，求为提供 20 dB 的干扰余裕度所必需的处理增益。

解　题

利用式(9.2.15)，求出处理增益$(W/R)_{dB} = 30$ dB，即 $W/R = L_c = 1000$。这表示在 $W/R = 1000$ 下，接收机中的平均干扰功率可以是期望信号功率 P_s 的 100 倍，这时还能维持可靠的通信。

编码扩频信号的性能

正如在第 8 章中指出的，当传输信息用二进制线性(分组或卷积)码编码时，在软判决解码器输出中的 SNR 随编码增益而提高，编码增益定义为：

$$\text{编码增益 } CG = R_c d_{min}^H \tag{9.2.16}$$

其中，R_c 是码率，d_{min}^H是该码的最小 Hamming 距离。因此，编码的效果就是通过编码增益而提高了干扰余裕度。这样，式(9.2.15)就修正为：

$$\left(\frac{P_J}{P_S}\right)_{dB} = \left(\frac{W}{R}\right)_{dB} + (CG)_{dB} - \left(\frac{\mathscr{E}_b}{J_0}\right)_{dB} \tag{9.2.17}$$

9.2.3　DS 扩频信号的两个应用方面

在这个小节中，要简要叙述一下 DS 扩频信号在两个方面的应用。首先考虑信号在很低功率传输中的应用，这样的情况可使某一收听者在试图检测出信号的存在时遇到很大困难。第二种应用是在多址无线通信中。

低可检测性的信号传输

在这类应用中，载有信息的信号相对于接收机前端产生的背景信道噪声和热噪声而言是在很低的功率电平下传输的。如果 DS 扩频信号占有的带宽为 W，而加性噪声的功率谱密度

是 N_0W/Hz,那么在带宽 W 内的平均噪声功率是 $P_N = WN_0$。

在计划的目标接收机中,平均接收信号功率是 P_R。如果我们希望将信号的存在从与目标接收机邻近的接收机隐藏起来,就可以使输信号的功率满足 $P_R/P_N << 1$。借助于处理增益和编码增益,这个目标接收机就能够从背景噪声中恢复出这个微弱的载有信息的信号。然而,任何其他接收机都没有任何有关 PN 码的知识,因此不可能利用这个处理增益和编码增益,结果载有信息的信号很难被检测到。我们认为这个传输信号有一个**低的被截获的概**率(low probability of being intercepted. LPI),并将它称为 LPI 信号。

在 9.2.2 节中,给出的差错概率也能用在目标接收机的 LPI 信号的解调和解码上。

解说题

解说题 9.2　[DS 扩频系统设计]

设计一个 DS 扩频信号,使其对一个 AWGN 信道在目标接收机的功率比是 $P_R/P_N = 0.01$。对于所接受的性能,期望值是$\mathcal{E}_b/N_0$。为达到$\mathcal{E}_b/N_0 = 10$,求所需处理增益的最小值。

题　解

可以将$\mathcal{E}_b/N_0$ 写成:

$$\frac{\mathcal{E}_b}{N_0} = \frac{P_R T_b}{N_0} = \frac{P_R L_c T_c}{N_0} = \left(\frac{P_R}{WN_0}\right) L_c = \left(\frac{P_R}{P_N}\right) L_c \qquad (9.2.18)$$

因为$\mathcal{E}_b/N_0 = 10$, $P_R/P_N = 10^{-2}$,所以所需处理的增益是 $L_c = 1000$。

码分多址

从 DS 扩频信号通过处理增益和编码增益所获得的性能上的增强,有可能用来在同一个信道带宽内拥有多个 DS 扩频信号,只要每个信号都有它自己的伪随机(特征)序列。因此,就有可能几个用户在同一个信道带宽内同时传输信息。这种在一条公共信道带宽内,每对发送/接收机用户都有它自己的不同传输特征码的数字通信类型,称为**码分多址**(code division multiple access, CDMA)。

在数字蜂窝通信系统中,一个基站利用 N_u 个正交 PN 序列传送信号到 N_u 个移动接收机,其中每个 PN 序列对应于一个目标接收机。这 N_u 个信号在传输时都是精确同步的,使得它们都以同步状态到达每个移动接收机。这样,由于这 N_u 个 PN 序列的正交性,每个目标接收机都能解调出它自己的信号而不会受到来自共享同一带宽的其他传输信号的干扰。然而,在从移动发射机到基站传输(上行线路)的信号中,这种同步型式不能保持。在基站的每个 DS 扩频信号的解调中,从该信道的其他并存用户来的信号就作为加性干扰出现。现在来确定在 CDMA 系统中能够容纳的并存信号的数目。假设基站的全部信号都有相同的平均功率。在许多实际系统中,来自每个用户的接收信号的功率在基站都受到监控,利用控制信道在全部并存用户上实施功率控制,并通知用户是否应提高或降低它们的功率电平。有了这样的功率控制之后,如果存在 N_u 个用户,那么在某一接收机,信号对噪声干扰的功率比是:

$$\frac{P_S}{P_N} = \frac{P_S}{(N_u - 1)P_S} = \frac{1}{N_u - 1} \qquad (9.2.19)$$

根据这一关系就可以确定同时容纳的用户数。

在确定信道可同时存在的最大用户数时,已经隐含着假设各个用户所用的伪随机码序列是正交的,而且从其他用户来的干扰仅加在某一功率基上。然而,要实现在 N_u 个用户之间伪随机序列的正交性,一般是很困难的,尤其是当 N_u 很大时更是如此。事实上,设计一批具有很好相关性质的伪随机序列就是一个重要的课题,并在专业技术文献中受到极大的关注。在 9.3 节将对此进行简要讨论。

解说题

解说题 9.3 [CDMA 中的最大用户数]

假设在某个 CDMA 系统中某用户所期望的性能水平是在 $\mathscr{E}_b/J_0 = 10$ 下达到的,若带宽对比特率之比是 100,编码增益是 6 dB,求在该 CDMA 系统中可同时容纳的最大用户数。

题　解

根据式(9.2.17)中给出的基本关系,有:

$$\left(\frac{P_N}{P_S}\right)_{dB} = \left(\frac{W}{R}\right)_{dB} + (CG)_{dB} - \left(\frac{\mathscr{E}_b}{J_0}\right)_{dB}$$
$$= 20 + 6 - 10 = 16\ dB$$

因此,

$$\frac{1}{N_u - 1} = \frac{P_S}{P_N} = \frac{1}{40}$$

所以

$$N_u = 41 \text{ 个用户}$$

解说题

解说题 9.4 [DS 扩频仿真]

这个题的目的是通过 Monte Carlo 仿真说明 DS 扩频信号在抑制正弦干扰方面的有效性。待仿真的系统的方框图如图 9.5 所示。

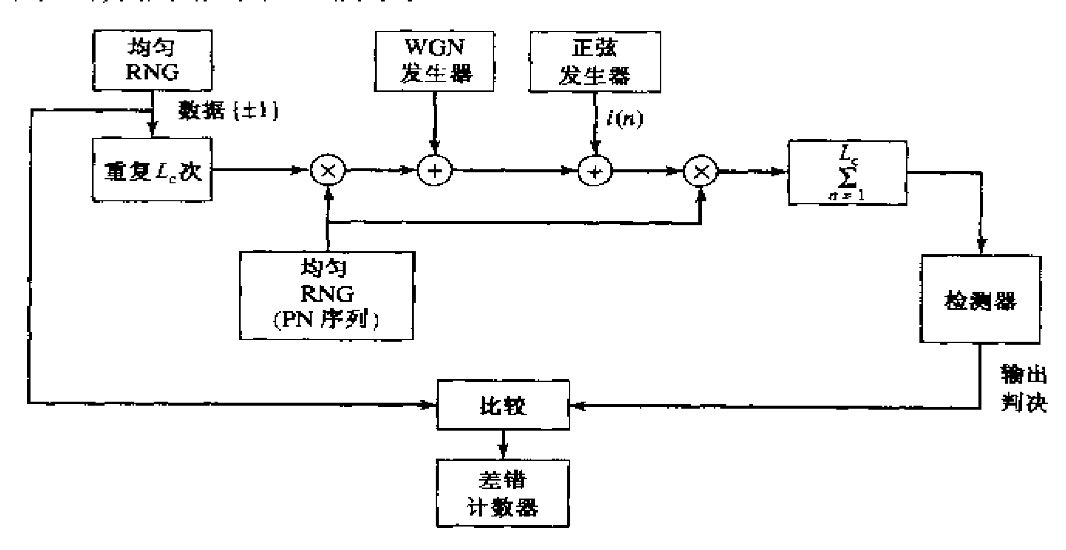


图 9.5　用于 Monte Carlo 仿真的 DS 扩频系统的模型

题 解

用一个均匀随机数发生器(RNG)产生某个二进制信息符号(±1)的序列。每个信息比特重复 L_c 次,L_c 相应于每个信息比特的 PN 码片数。所得到的序列(其中包含每比特的 L_c 个重复数)乘以由另一个均匀 RNG 产生的 PN 序列 $c(n)$。然后,将方差为 $\sigma^2 = N_0/2$ 的高斯白噪声和形式为:

$$i(n) = A\sin(\omega_0 n)$$

的正弦干扰加到这个乘积序列上,其中 $0 < \omega_0 < \pi$,正弦的幅度选为 $A < L_c$。解调器完成与 PN 序列的互相关,并在构成每信息比特的 L_c 个信号样本上求和(积分)。相加器的输出再馈给检测器,它将这个信号与阈值零进行比较,并判决传输的比特是 +1 还是 -1。差错计数器计出由检测器产生的差错数。对于 $L_c = 20$,在 3 种不同的正弦干扰幅值下所得的 Monte Carlo 仿真结果如图 9.6 所示。图中还给出了当除掉这个正弦干扰时所测得的误码率。在这些仿真过程中,加性噪声的方差都保持不变,而在每次仿真运行中,所期望的信号电平都加权到能够实现所需的 SNR。

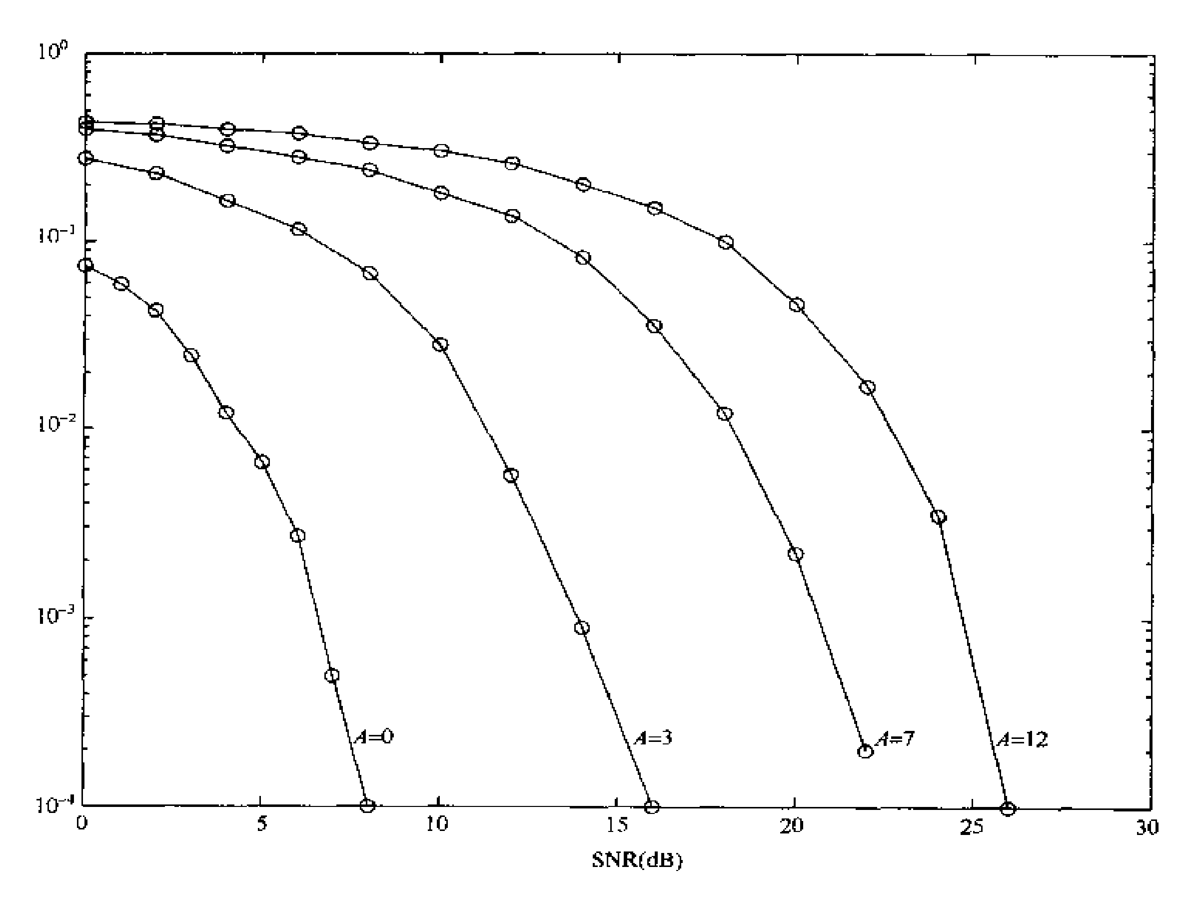


图 9.6 从 Monte Carlo 仿真所得的系统的误码率性能

该仿真程序的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 9.4.
echo on
Lc=20;                                  % number of chips per bit
```

```
A1=3;                              % amplitude of the first sinusoidal interference
A2=7;                              % amplitude of the second sinusoidal interference
A3=12;                             % amplitude of the third sinusoidal interference
A4=0;                              % fourth case: no interference
w0=1;                              % frequency of the sinusoidal interference in radians
SNRindB=0:2:30;
for i=1:length(SNRindB),
  % measured error rates
  smld_err_prb1(i)=ss_Pe94(SNRindB(i),Lc,A1,w0);
  smld_err_prb2(i)=ss_Pe94(SNRindB(i),Lc,A2,w0);
  smld_err_prb3(i)=ss_Pe94(SNRindB(i),Lc,A3,w0);
  echo off ;
end;
echo on ;
SNRindB4=0:1:8;
for i=1:length(SNRindB4),
  % measured error rate when there is no interference
  smld_err_prb4(i)=ss_Pe94(SNRindB4(i),Lc,A4,w0);
  echo off ;
end;
echo on ;
% Plotting commands follow.
```

m 文件

```
function [p]=ss_Pe94(snr_in_dB, Lc, A, w0)
% [p]=ss_Pe94(snr_in_dB, Lc, A, w0)
%               SS_PE94  finds the measured error rate. The function
%               that returns the measured probability of error for the given value of
%               the snr_in_dB, Lc, A and w0.
snr=10^(snr_in_dB/10);
sgma=1;                            % Noise standard deviation is fixed.
Eb=2*sgma^2*snr;                   % signal level required to achieve the given
                                   % signal-to-noise ratio
E_chip=Eb/Lc;                      % energy per chip
N=10000;                           % number of bits transmitted
% The generation of the data, noise, interference, decoding process and error
% counting is performed all together in order to decrease the run time of the
% program. This is accomplished by avoiding very large sized vectors.
num_of_err=0;
for i=1:N,
  % Generate the next data bit.
  temp=rand;
  if (temp<0.5),
    data=-1;
  else
    data=1;
  end;
  % Repeat it Lc times, i.e. divide it into chips.
  for j=1:Lc,
    repeated_data(j)=data;
  end;
  % pn sequence for the duration of the bit is generated next.
  for j=1:Lc,
    temp=rand;
    if (temp<0.5),
      pn_seq(j)=-1;
    else
      pn_seq(j)=1;
    end;
```

```
  end;
  % the transmitted signal is
  trans_sig=sqrt(E_chip)*repeated_data.*pn_seq;
  % AWGN with variance sgma^2
  noise=sgma*randn(1,Lc);
  % interference
  n=(i-1)*Lc+1:i*Lc;
  interference=A*sin(w0*n);
  % received signal
  rec_sig=trans_sig+noise+interference;
  % Determine the decision variable from the received signal.
  temp=rec_sig.*pn_seq;
  decision_variable=sum(temp);
  % making decision
  if (decision_variable<0),
    decision=-1;
  else
    decision=1;
  end;
  % If it is an error, increment the error counter.
  if (decision~=data),
    num_of_err=num_of_err+1;
  end;
end;
% then the measured error probability is
p=num_of_err/N;
```

9.3 PN序列的产生

一个伪随机(或PN)序列是一个1和0的码序列,它的自相关函数具有与白噪声自相关函数相似的性质。这一节要简要叙述某PN序列的构成以及它们的自相关和互相关性质。

到现在为止,最为大家熟知的二进制PN码的序列是最大长度移位寄存器序列。一个最大长度移位寄存器序列(或简称为m序列)的长度为$L=2^m-1$比特,并由一个m级的带有线性反馈的移位寄存器产生,如图9.7所示。这个序列是周期的,周期为L。每个周期内有2^{m-1}个1和$2^{m-1}-1$个0的序列。表9.1列出了产生最大长度序列的移位寄存器的连接。

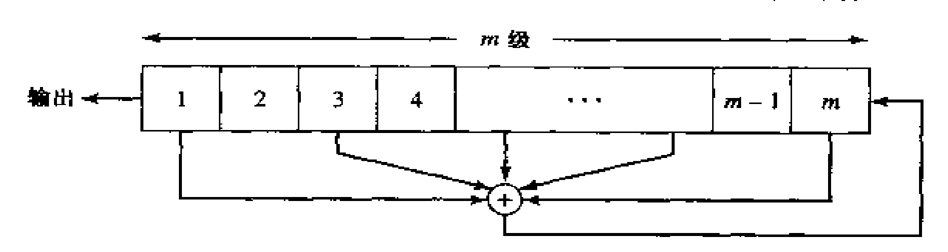


图9.7　具有线性反馈的一般m级移位寄存器

表9.1　产生最大长度序列的移位寄存器连接法

m	接至模2加法器的级	m	接至模2加法器的级	m	接至模2加法器的级
2	1,2	13	1,10,11,13	24	1,18,23,24
3	1,3	14	1,5,9,14	25	1,23

（续表）

m	接至模 2 加法器的级	m	接至模 2 加法器的级	m	接至模 2 加法器的级
4	1,4	15	1,15	26	1,21,25,26
5	1,4	16	1,5,14,16	27	1,23,26,27
6	1,6	17	1,15	28	1,26
7	1,7	18	1,12	29	1,28
8	1,5,6,7	19	1,15,18,19	30	1,8,29,30
9	1,6	20	1,18	31	1,29
10	1,8	21	1,20	32	1,11,31,32
11	1,10	22	1,22	33	1,21
12	1,7,9,12	23	1,19	34	1,8,33,34

在 DS 扩频应用中，码元为{0,1}的二进制序列映射为码元为{−1，+1}的相应二进制序列。这个等效的码元为{−1，+1}的序列{c_n}称为**双极性序列**(bipolar sequence)。

一个周期 PN 序列的重要特性是其自相关函数，双极性序列{c_n}的自相关函数通常定义为：

$$R_c(m)=\sum_{n=1}^{L}c_nc_{n+m},\quad 0\leqslant m\leqslant L-1 \tag{9.3.1}$$

其中，L 是该序列的周期。因为序列{c_n}是周期的，周期为 L，所以自相关序列{$R_c(m)$}也是周期的，周期为 L。

在理想情况下，一个 PN 序列应该有一个自相关函数，其相关性质与白噪声的自相关性质是类似的。这就是说，对于{c_n}的理想自相关序列是 $R_c(0)=L$ 和 $R_c(m)=0,1\leqslant m\leqslant L-1$。在 m 序列情况下，自相关序列是：

$$R_c(m)=\begin{cases}L, & m=0\\ -1, & 1\leqslant m\leqslant L-1\end{cases} \tag{9.3.2}$$

对于长 m 序列，$R_c(m)$相对于峰值 $R_c(0)$的偏离值，即 $R_c(m)/R_c(0)=-1/L$ 是很小的，从实际的角度来看无关紧要。因此，当通过它们的自相关函数来看时，m 序列非常接近于理想 PN 序列。

在一些应用中，PN 序列的互相关性质和自相关性质具有同样的重要性。例如，在 CDMA 中，每个用户都分配到某一特定的 PN 序列。在理想情况下，在用户之间的这些 PN 序列应该是互为正交的，以使一个用户受到来自其他用户传输的干扰电平是零。然而，在实际中被不同用户使用的 PN 序列总是呈现某些相关性。

现在我们具体考虑这类 m 序列。已经知道，在相同周期的一对 m 序列之间的周期互相关函数可能有相当大的峰值。表 9.2 中列出了当 $3\leqslant m\leqslant 12$ 时，各对 m 序列之间周期互相关的峰值幅度 R_{max}；同时，在该表中还列出了当 $3\leqslant m\leqslant 12$ 时，长度为 $L=2^m-1$ 的 m 序列的数目。可以看到，长为 L 的 m 序列的数目随着 m 急剧增加；同时还可以看到，对于大多数序列来说，互相关函数的峰值幅度 R_{max} 是自相关函数峰值的一个大的百分比。因此，m 序列对 CDMA 通信系统来说是不合适的。虽然有可能挑选出一小部分 m 序列，它具有比 R_{max} 相对小的互相关峰值，但是这一部分序列的数目还是太少，不足以在 CDMA 中应用。

表 9.2 m 序列和 Gold 序列的峰值互相关

m	$L=2^m-1$	m 序列			Gold 序列	
		序列数	R_{max}	$R_{max}/R(0)$	R_{max}	$R_{max}/R(0)$
3	7	2	5	0.71	5	0.71
4	15	2	9	0.60	9	0.60
5	31	6	11	0.35	9	0.29
6	63	6	23	0.36	17	0.27
7	127	18	41	0.32	17	0.13
8	255	16	95	0.37	33	0.13
9	511	48	113	0.22	33	0.06
10	1023	60	383	0.37	65	0.06
11	2047	176	287	0.14	65	0.03
12	4095	144	1407	0.34	129	0.03

Gold[5],[6]和 Kasami[7]已经研究出产生具有比 m 序列更好的周期互相关函数性质的 PN 序列的方法。Gold 序列是这样构成的:取一对专门挑选的 m 序列,称之为**优选 m 序列**(preferred m-sequences),并将其中一个序列相对于另一个序列做 L 次循环移位,对每次移位后按模 2 相加。据此,L Gold 序列的产生如图 9.8 所示。对于大的 L 和奇数 m,任意一对 Gold 序列之间的互相关函数的最大值 $R_{max}=\sqrt{2L}$;对于偶数 m,$R_{max}=\sqrt{L}$。

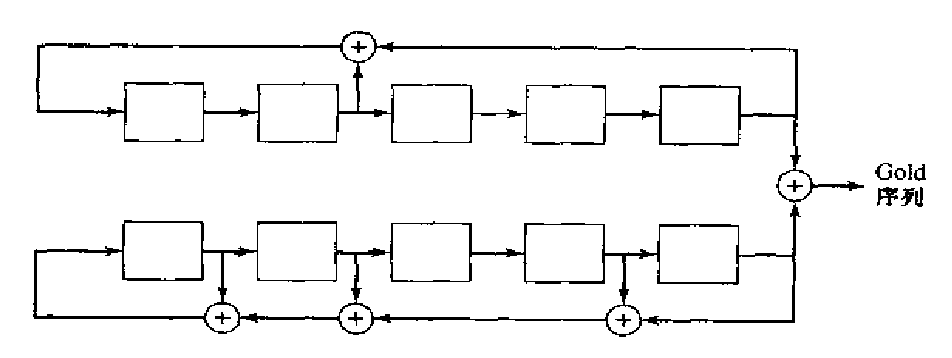


图 9.8 长度为 31 的 Gold 序列的产生

Kasami[7]提出一种通过抽取某一 m 序列构造 PN 序列的方法。在 Kasami 的构造方法中,一个 m 序列中每隔$(2^{m/2}+1)$个比特被抽取出。这种构造方法产生比 Gold 序列更少的一组 PN 序列,但是它们的最大互相关值是 $R_{max}=\sqrt{L}$。

用任意一对长度为 L 的二进制序列之间的最大互相关的某一已知低界,对 Gold 序列和 Kasami 序列的互相关函数的峰值进行比较是很有意思的。已知周期为 L 的 N 组序列,它们的最大互相关的低界是:

$$R_{max} \geqslant L\sqrt{\frac{N-1}{NL-1}} \tag{9.3.3}$$

对于大的 L 和 N 值,该式可以近似为 $R_{max}\geqslant\sqrt{L}$。因此可见,Kasami 序列满足这个低界,所以它们是最佳的。另外,m 为奇数的 Gold 序列有 $R_{max}=\sqrt{2L}$,所以它们稍微是次最佳的。

除了 Gold 序列和 Kasami 序列以外，还有其他一些二进制序列也适合于在 CDMA 中应用。有兴趣的读者可参阅文献 Scholtz[8]和文献 Sarwate and Pursley[9]。

解说题

解说题 9.5　[产生 Gold 序列]

产生 $L=31$ 的 Gold 序列，它由取如图 9.8 所示的两个移位寄存器输出的模 2 相加而产生。

题　解

完成这个计算的 MATLAB 脚本如下所示。产生的 31 个序列如表 9.3 所示。这些序列互相关的最大值是 $R_{max}=9$。

表 9.3　解说题 9.5 中的 Gold 序列表

0	1	1	1	0	0	0	0	1	0	0	0	0	1	1	0	0	1	0	0	1	0	1	1	1	1	0	0	0	0	0
0	1	0	1	1	0	1	1	1	0	0	0	1	0	0	0	0	0	1	0	0	1	0	0	0	1	1	0	0	0	1
0	0	0	0	1	1	0	1	1	0	0	1	0	1	0	0	1	1	1	1	1	0	1	1	0	0	1	0	0	1	1
1	0	1	0	0	0	0	1	1	0	1	0	1	1	0	1	0	1	0	0	0	1	0	1	1	0	1	0	1	1	1
1	1	1	1	1	0	0	1	1	1	0	1	1	1	1	0	0	0	1	1	1	0	0	0	1	0	1	1	1	1	0
0	1	0	0	1	0	0	1	0	0	1	1	1	0	0	0	1	1	0	0	0	0	1	0	1	0	0	1	1	0	0
0	0	1	0	1	0	0	0	1	1	1	1	0	1	0	1	0	0	1	1	0	1	1	0	1	1	0	1	0	0	1
1	1	1	0	1	0	1	1	0	1	1	0	1	1	1	0	1	1	0	1	1	1	1	0	0	1	0	0	0	1	1
0	1	1	0	1	1	0	0	0	1	0	1	1	0	0	1	0	0	0	0	1	1	1	1	0	1	1	0	1	1	0
0	1	1	0	0	0	1	0	0	0	1	1	0	1	1	0	1	0	1	0	1	1	0	1	0	0	1	1	1	0	1
0	1	1	1	1	1	1	0	1	1	1	0	1	0	0	1	1	1	1	0	1	0	0	1	1	0	0	1	0	1	1
0	1	0	0	0	1	1	1	0	1	0	1	0	1	1	1	0	1	1	0	0	0	0	0	1	1	0	0	1	1	1
0	0	1	1	0	1	0	0	0	0	1	0	1	0	1	0	0	1	1	1	0	0	1	0	0	1	1	1	1	1	1
1	1	0	1	0	0	1	0	1	1	0	1	0	0	0	0	0	1	0	1	0	1	1	1	0	0	0	1	1	1	1
0	0	0	1	1	1	1	1	0	0	1	0	0	1	0	0	0	0	0	1	1	1	0	1	1	1	0	1	1	1	0
1	0	0	0	0	1	0	0	1	1	0	0	1	1	0	0	1	0	0	0	1	0	0	0	0	1	0	1	1	0	1
1	0	1	1	0	0	1	1	0	0	0	1	1	1	0	1	1	0	1	0	0	0	1	1	0	1	0	1	0	1	0
1	1	0	1	1	1	0	0	1	0	1	1	1	1	1	1	1	1	1	1	0	1	0	1	0	1	0	0	1	0	0
0	0	0	0	0	0	1	1	1	1	1	1	1	0	1	1	0	1	0	1	1	0	0	1	0	1	1	1	0	0	0
1	0	1	1	1	1	0	1	0	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1
1	1	0	0	0	0	0	0	0	1	1	0	0	0	0	0	1	0	1	1	0	0	0	1	1	1	1	0	0	1	0
0	0	1	1	1	0	1	0	0	1	0	0	0	1	0	1	1	1	0	1	0	0	0	0	0	0	1	0	1	0	0
1	1	0	0	1	1	1	0	0	0	0	0	1	1	1	1	0	0	0	1	0	0	1	1	1	0	1	1	0	0	1
0	0	1	0	0	1	1	0	1	0	0	1	1	0	1	0	1	0	0	1	0	1	0	0	1	0	0	0	0	1	0
1	1	1	1	0	1	1	1	1	0	1	1	0	0	0	1	1	0	0	1	1	0	1	0	1	1	1	0	1	0	1
0	1	0	1	0	1	0	1	1	1	1	0	0	1	1	1	1	0	0	0	0	1	1	0	0	0	1	1	0	1	0
0	0	0	1	0	0	0	1	0	1	0	0	1	0	1	1	1	0	1	1	1	1	1	1	1	0	0	0	1	0	1
1	0	0	1	1	0	0	0	0	0	0	1	0	0	1	1	1	1	0	0	1	1	0	0	1	1	1	1	0	1	1
1	0	0	0	1	0	1	0	1	0	1	0	0	0	1	1	0	0	1	0	1	0	1	0	0	0	0	0	1	1	0
1	0	1	0	1	1	1	1	1	1	0	0	0	0	1	0	1	1	1	0	0	1	1	1	1	1	1	1	1	0	0
1	1	1	0	0	1	0	1	0	0	0	0	0	0	0	1	0	1	1	1	1	1	0	0	0	0	0	1	0	0	0

m 文件

```
% MATLAB script for Illustrative Problem 9.5.
echo on
% First determine the maximal length shift-register sequences.
% Assume the initial shift-register content as "00001".
connections1=[1 0 1 0 0];
connections2=[1 1 1 0 1];
sequence1=ss_mlsrs(connections1);
sequence2=ss_mlsrs(connections2);
% Cyclically shift the second sequence and add it to the first one.
L=2^length(connections1)-1;;
for shift_amount=0:L-1,
  temp=[sequence2(shift_amount+1:L) sequence2(1:shift_amount)];
  gold_seq(shift_amount+1,:)=(sequence1+temp) - floor((sequence1+temp)./2).*2;
  echo off ;
end;
echo on ;
% Find the max value of the cross-correlation for these sequences.
max_cross_corr=0;
for i=1:L-1,
  for j=i+1:L,
    % equivalent sequences
    c1=2*gold_seq(i,:)-1;
    c2=2*gold_seq(j,:)-1;
    for m=0:L-1,
      shifted_c2=[c2(m+1:L) c2(1:m)];
      corr=abs(sum(c1.*shifted_c2));
      if (corr>max_cross_corr),
        max_cross_corr=corr;
      end;
      echo off ;
    end;
  end;
end;
% Note that max_cross_correlation turns out to be 9 in this example.
```

m 文件

```
function [seq]=ss_mlsrs(connections);
% [seq]=ss_mlsrs(connections)
%           SS_MLSRS  generates the maximal length shift-register sequence when the
%           shift-register connections are given as input to the function. A "zero"
%           means not connected, whereas a "one" represents a connection.
m=length(connections);
L=2^m-1;                          % length of the shift register sequence requested
registers=[zeros(1,m-1) 1];       % initial register contents
seq(1)=registers(m);              % first element of the sequence
for i=2:L,
  new_reg_cont(1)=connections(1)*seq(i-1);
  for j=2:m,
    new_reg_cont(j)=registers(j-1)+connections(j)*seq(i-1);
  end;
  registers=new_reg_cont;         % current register contents
  seq(i)=registers(m);            % the next element of the sequence
end;
```

9.4 跳频扩频

在跳频(FH)扩频中,将可利用的信道带宽 W 划分成大量非重叠的频率间隙,在任何信号区间内,传输信号占据一个或多个可用的频率间隙。在每个信号区间内,频率间隙(一个或多个)的选取是按照来自某一 PN 发生器的输出伪随机地确定的。

图 9.9 给出了一个 FH 扩频系统发射端和接收端的方框图。调制既可用二进制也可以用 M 元 FSK(MFSK)。例如,如果采用二进制 FSK,那么调制器就选取两个频率中的一个(比如 f_0 或 f_1)对应于传输一个 0 或 1。然后,再将这个二进制的 FSK 信号在频率上搬移一个量,这个量由来自某个 PN 发生器的输出序列决定,用这个 PN 发生器选择某一被频率合成器同步的频率 f_c。这个频率被搬移了的信号在信道上传输。例如,通过从 PN 发生器取 m 个比特,就可以给出 2^m-1 个可能的载波频率。图 9.10 说明了一个 FH 信号的示意图。

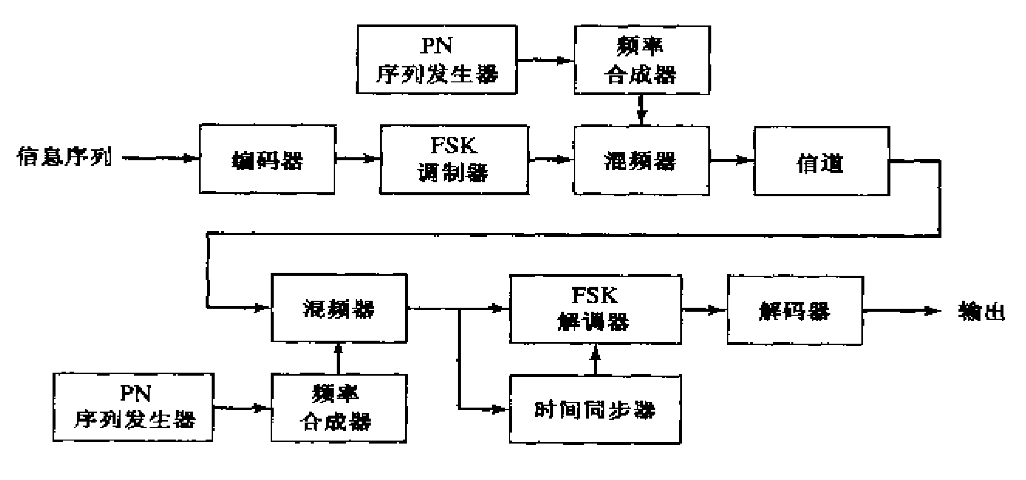


图 9.9　FH 扩频系统的方框图

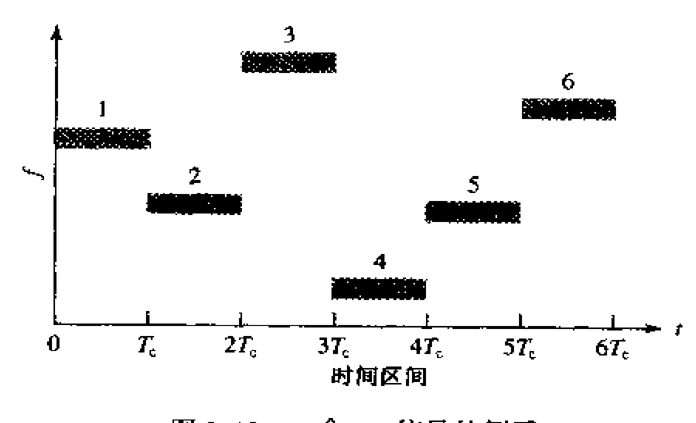


图 9.10　一个 FH 信号的例子

在接收端有一个完全相同的 PN 序列发生器,它是与接收信号同步的,用来控制频率合成器的输出。通过频率合成器输出与接收信号混频,然后由发射端引入的伪随机频率搬移在解调中被移去。这样即可利用 FSK 解调出最后所得信号。用于维持 PN 序列发生器与 FH 接收信号同步的信号通常是从接收信号中提取出来的。

虽然一般来说二进制 PSK 调制比二进制 FSK 产生了更好的性能,但是要保持用于跳频模式的频率合成中的相位相干很困难。并且,当信号在一个宽的带宽上从一个频率跳到另一个频率时,在信道上信号的传播过程中保持相位相干也很困难。因此,在 FH 扩频系统中,一般都采用非相干解调的 FSK 调制。

频率跳变率 R_h 可以选为等于符号率,低于符号率或高于符号率。如果 R_h 等于或低于符号率,这个 FH 系统称为**慢跳频**(slow-hopping)系统;如果 R_h 高于符号率,也就是每个符号区间内有多次跳频,则称这个 FH 系统为**快跳频**(fast-hopping)系统。我们仅考虑跳变率等于符号率的情况。

9.4.1　FH 信号的差错概率

现考虑一个 FH 系统,其中用二进制 FSK 传输数字信息。频率跳变是每比特跳一次。解调和检测都是非相干的。在 AWGN 信道上,这样的系统的差错概率是:

$$P_b = \frac{1}{2}e^{-\mathscr{E}_b/(2N_0)} \tag{9.4.1}$$

如果干扰是一个宽带信号,或者是覆盖整个 FH 带宽 W 内的平坦频谱的干扰,上述结果也适用。在这种情况下,N_0 要用 $N_0 + J_0$ 代替,其中 J_0 是干扰的谱密度。

和在 DS 扩频系统中的情况一样,每比特能量$\mathscr{E}_b$ 可以表示为$\mathscr{E}_b = P_S T_b = P_S/R$,其中 P_S 是平均信号功率,R 是比特率。类似地,也有 $J_0 = P_J/W$,其中 P_J 是宽带干扰的平均功率,W 为可用的信道带宽。因此,假设 $J_0 >> N_0$,就能将 SNR 表示为:

$$\frac{\mathscr{E}_b}{J_0} = \frac{W/R}{P_J/P_S} \tag{9.4.2}$$

其中,W/R 是处理增益,P_J/P_S 是 FH 扩频信号的干扰余裕度。

慢 FH 扩频系统特别容易受到部分频带干扰攻击,这一干扰既可以来自专门的干扰台,也可以来自 FH CDMA 系统。具体来说,假设这个部分频带干扰用一个零均值的高斯随机过程来建模,这个过程在总带宽 W 的某一部分具有平坦的功率谱密度,而在这个频带以外是零。在功率谱密度为非零的区域,它的值是$S_J(f) = J_0/\alpha$,其中 $0 < \alpha \leqslant 1$。换句话说,干扰平均功率 P_J 假设是常数,而 α 是被干扰所占的频带份额。

假设部分频带干扰来自某一干扰台,干扰台挑选 α,以使其对通信系统的效果是最佳的。在采用二进制 FSK 调制和非相干检测的未编码慢跳频系统中,传输频率在频带 W 内选成了均匀概率的。这样,接收信号将以概率 α 被干扰,而以概率 $1-\alpha$ 不被干扰。当被干扰时,差错概率是$(1/2)\exp(-\alpha\rho_b/2)$,而当不被干扰时,信号检测假定是无差错的,其中 $\rho_b \equiv \mathscr{E}_b/J_0$。因此,平均差错概率是:

$$\begin{aligned} P_2(\alpha) &= \frac{\alpha}{2}\exp\left(-\frac{\alpha\rho_b}{2}\right) \\ &= \frac{\alpha}{2}\exp\left(-\frac{\alpha W/R}{2P_J/P_S}\right) \end{aligned} \tag{9.4.3}$$

图 9.11 给出了对于几个 α 值,作为 SNR_{ρ_b} 的函数的误码率。干扰台假设是以对差错概率最大的优化策略选取 α 的,通过对 $P_2(\alpha)$微分并解出使 $P_2(\alpha)$最大的 α 值,就能求得干扰台的最好 α 的取值为:

$$\alpha^* = \begin{cases} 2/\rho_b, & \rho_b \geqslant 2 \\ 1, & \rho_b < 2 \end{cases} \tag{9.4.4}$$

对于最坏情况下的部分频带干扰,相应的差错概率是:

$$P_2 = \begin{cases} e^{-1}/\rho_b, & \rho_b \geqslant 2 \\ \dfrac{1}{2}e^{-\rho_b/2}, & \rho_b < 2 \end{cases} \tag{9.4.5}$$

如图 9.11 所示。在如式(9.4.3)给出的全频带干扰下,其差错率随$\mathscr{E}_b/J_0$ 增加而呈指数下降,而在最坏部分频带干扰下仅成反比减小。

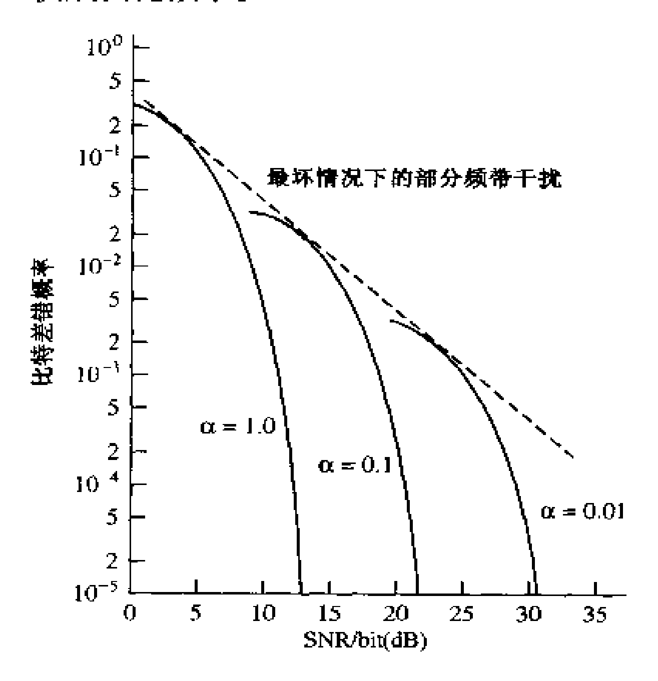


图 9.11 部分频带干扰时的二进制 FSK 性能

解说题

解说题 9.6 [FH 系统仿真]

通过 Monte Garlo 仿真,说明一个使用二进制 FSK 并受到最坏部分频带干扰破坏的 FH 数字通信系统的性能。待仿真的系统的方框图见图 9.12。

题 解

用均匀随机数发生器(RNG)产生某个二进制信息序列,它作为 FSK 调制器的输入,FSK 调制器的输出受到概率 α 的加性高斯噪声污损,$0<\alpha\leqslant 1$。用第二个均匀 RNG 来决定这个加性高斯噪声何时污损信号以及何时不污损信号。在有噪声时,假定在传输为 0 的情况下,检测器的输入是:

$$r_1 = (\sqrt{\mathscr{E}_b}\cos\phi + n_{1c}{}^{①})^2 + (\sqrt{\mathscr{E}_b}\sin\phi + n_{1s}{}^{②})^2$$
$$r_2 = n_{2c}^2 + n_{2s}^2$$

① 原著此处为 n_{cr} —译者注。

② 原著此处为 n_s。—译者注。

其中,ϕ 是信道相移,n_{1c},n_{1s},n_{2c}和 n_{2s}代表加性噪声分量。在没有噪声时有:

$$r_1 = \mathcal{E}_b, \qquad r_2 = 0$$

所以在检测器中没有差错发生。每个噪声分量的方差是 $\sigma^2 = \alpha J_0/2$,其中 α 由式(9.4.4)给出。为简单起见,置 $\phi=0$ 并将 J_0 归一化到 1,那么 $\rho_b = \mathcal{E}_b/J_0 = \mathcal{E}_b$。因为 $\sigma^2 = J_0/2\alpha$,$\alpha = 2/\rho_b$,接着就能得出,在部分频带干扰存在的情况下,$\sigma^2 = \mathcal{E}_b/4$,$\alpha = 2/\mathcal{E}_b$,这里把$\mathcal{E}_b$ 限制到$\mathcal{E}_b \geqslant 2$。图 9.13给出了由 Monte Garlo 仿真得出的误码率。图中还示意了由式(9.4.5)给出的差错概率的理论值。

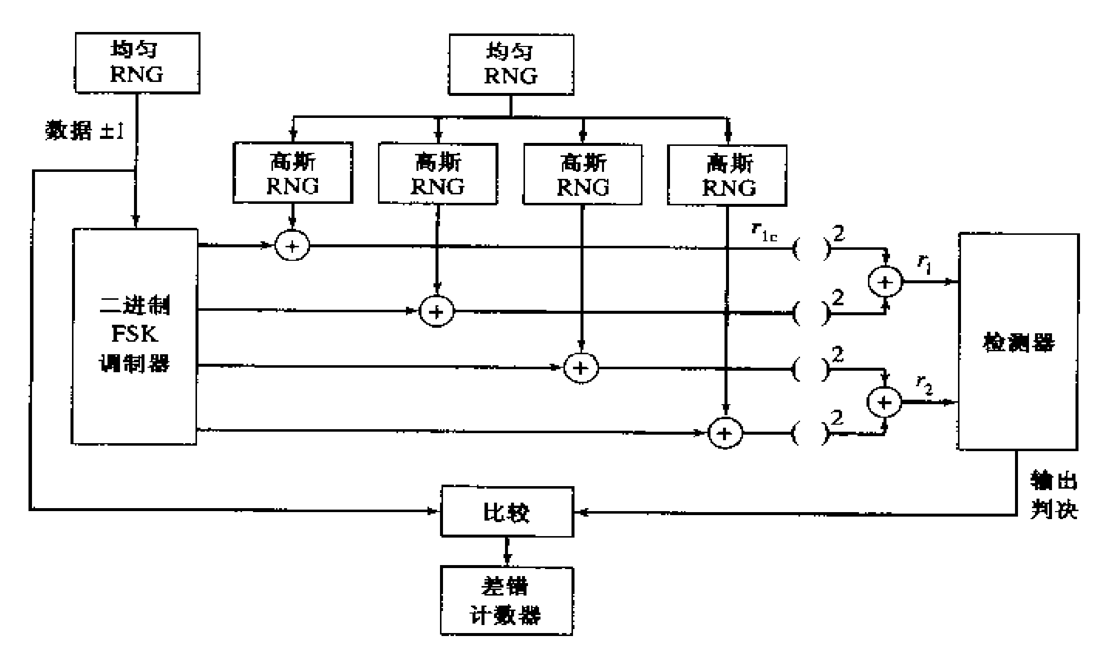


图 9.12 Monte Garlo 仿真:部分频带干扰下的二进制 FSK 系统的模型

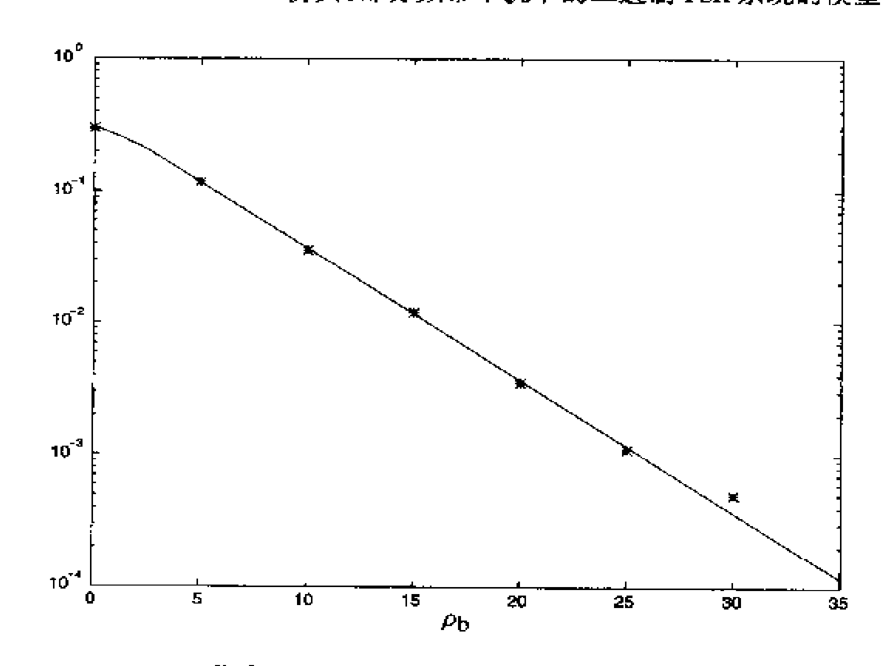


图 9.13 Monte Garlo 仿真:在部分频带干扰下,FH 二进制 FSK 系统的误码率性能

该仿真程序的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB script for Illustrative Problem 9.6.
echo on
rho_b1=0:5:35;                          % rho in dB for the simulated error rate
rho_b2=0:0.1:35;                        % rho in dB for theoretical error rate computation
for i=1:length(rho_b1),
  smld_err_prb(i)=ss_pe96(rho_b1(i));   % simulated error rate
  echo off ;
end;
echo on ;
for i=1:length(rho_b2),
  temp=10^(rho_b2(i)/10);
  if (temp>2)
    theo_err_rate(i)=1/(exp(1)*temp);   % theoretical error rate if rho>2
  else
    theo_err_rate(i)=(1/2)*exp(-temp/2);% theoretical error rate if rho<2
  end;
  echo off ;
end;
echo on ;
% Plotting commands follow.
```

m 文件

```
function [p]=ss_Pe96(rho_in_dB)
% [p]=ss_Pe96(rho_in_dB)
%               SS_PE96  finds the measured error rate. The value of
%               signal per interference ratio in dB is given as an
%               input to the function.
rho=10^(rho_in_dB/10);
Eb=rho;                                 % energy per bit
if (rho>2),
  alpha=2/rho;                          % optimal alpha if rho>2
else
  alpha=1;                              % optimal alpha if rho<2
end;
sgma=sqrt(1/(2*alpha));                 % noise standard deviation
N=10000;                                % number of bits transmitted
% generation of the data sequence
for i=1:N,
  temp=rand;
  if (temp<0.5)
    data(i)=1;
  else
    data(i)=0,
  end;
end;
% Find the received signals.
for i=1:N,
  % the transmitted signal
  if (data(i)==0),
    r1c(i)=sqrt(Eb);
    r1s(i)=0;
    r2c(i)=0;
    r2s(i)=0;
```

```
  else
    r1c(i)=0;
    r1s(i)=0;
    r2c(i)=sqrt(Eb);
    r2s(i)=0;
  end;
  % The received signal is found by adding noise with probability alpha.
  if (rand<alpha),
    r1c(i)=r1c(i)+gngauss(sgma);
    r1s(i)=r1s(i)+gngauss(sgma);
    r2c(i)=r2c(i)+gngauss(sgma);
    r2s(i)=r2s(i)+gngauss(sgma);
  end;
end;
% Make the decisions and count the number of errors made.
num_of_err=0;
for i=1:N,
  r1=r1c(i)^2+r1s(i)^2;          % first decision variable
  r2=r2c(i)^2+r2s(i)^2;                % second decision variable
  % Decision is made next.
  if (r1>r2),
    decis=0;
  else
    decis=1;
  end;
  % Increment the counter if this is an error.
  if (decis~=data(i)),
    num_of_err=num_of_err+1;
  end;
end;
% measured bit error rate is then
p=num_of_err/N;
```

9.4.2 利用信号分集解决部分频带干扰问题

正如在前一节中所讨论的,受到部分频带干扰侵扰的 FH 系统的性能很差。例如,该系统的差错概率若能达到 10^{-6},则在存在最坏情况的干扰时,检测器所需的 SNR 约为 60 dB。作为比较,在没有部分频带干扰时,在一个 AWGN 信道中所需的 SNR 约为 10 dB。结果,由于部分频带干扰的存在,在 SNR 上的损失竟有约 50 dB,这实在是太高了。

减小部分频带干扰对 FH 扩频系统的影响的方法是:通过信号分集;也就是说,将同一信息比特在多次频率跳变上传输,将多次传输的信号加权并在检测器输入端相加。具体来说,假设每个信息比特在两个相继的频率跳变上传输,这个系统称为**双分集系统**(dual diversity system)。这时,假设传输的是一个 0,或者在相加器的两个输入都受到干扰破坏,或者两个传输信号中的一个受到干扰破坏,或者两个传输信号都未受到干扰破坏。

假设相加器知道干扰的电平,由此可以形成组合判决变量:

$$x = w_1 r_{11} + w_2 r_{12}$$

$$y = w_1 r_{21} + w_2 r_{22} \tag{9.4.6}$$

其中,r_{11} 和 r_{21} 是对第一个传输信号平方律检测的两个输出,r_{12} 和 r_{22} 是由第二个传输信号的平方律检测的两个输出。加权系数 w_1 和 w_2 设置为 $1/\sigma^2$,σ^2 是加性噪声加干扰后的方差,因此

当 σ^2 较大时就对应于干扰存在的情况，加在接收信号上的权系数就小。另外，当 σ^2 较小时就对应于没有干扰的情况，加在接收信号上的权系数就大。这样，相加器就将受到干扰破坏的接收信号分量去加重。

从相加器来的两个分量 x 和 y 送给检测器，检测器就以较大的信号分量做出判决。

现在，具有双分集的 FH 信号的性能就由两个传输信号都受到干扰破坏的情况所决定。然而，这个事件的概率是正比于 α^2 的，远远小于 α。结果，对于最坏情况下部分频带干扰的差错概率就有如下形式：

$$P_2(2) = \frac{K_2}{\rho_b^2}, \qquad \rho_b > 2 \tag{9.4.7}$$

其中，K_2 是某一常数，$\rho_b = \mathcal{E}_b/J_0$。在这种情况下，双分集系统的差错概率随 SNR 的平方成反比下降。换句话说，在 SNR 上增加 10 倍(10 dB)就会得到在差错概率上降低 100 倍。这样，在双分集情况下，用大约 30 dB 的 SNR 就能实现 10^{-6} 的差错概率；作为对比，在没有分集时需要 60 dB 的 SNR，相差 1000 倍。

更为一般的情况是，若每个信息比特在 D 次频率跳变上传输，D 是分集的阶次，那么差错概率就有：

$$P_2(D) = \frac{K_D}{\rho_b^D}, \qquad \rho_b > 2 \tag{9.4.8}$$

其中，K_D 是某个常数。

因为以上讨论的信号分集是编码的一种简单形式(重复编码)，所以能采用最小 Hamming 距离等于 D 的一种码，并用平方律检测输出的软判决解码来代替每个信息比特重复传输 D 次，这就不足为奇了。

解说题

解说题 9.7　[在 FH 系统中的分集]

重做解说题 9.6 中 FH 系统的 Monte Garlo 仿真，但是现在采用双分集。

题　解

在没有干扰时，相加器中使用的权系数是置 $w = 10$，它对应于 $\sigma^2 = 0.1$，这可能是加性高斯噪声电平的典型值。另外，当有干扰时，权系数置为 $w = 1/\sigma^2 = 2/\mathcal{E}$，这里将 $\mathcal{E}$ 限制到 $\mathcal{E} \geqslant 4$。每次跳变的 SNR 是 $\mathcal{E}$，而在两次跳变中每比特的总能量是 $\mathcal{E}_b = 2\mathcal{E}$。因此，差错概率是作为 $\mathcal{E}_b/J_0$ 的函数画出的。这个 Monte Carlo 仿真结果如图 9.14 所示。该仿真程序的 MATLAB 脚本如下所示。

m 文件

```
% MATLAB scripit for Illustrative Problem 9.7.
echo on
rho_b=0:2:24;                              % rho in dB
for i=1:length(rho_b),
  smld_err_prb(i)=ss_Pe97(rho_b(i));       % simulated error rate
  echo off ;
end;
echo on ;
% Plotting commands follow.
```

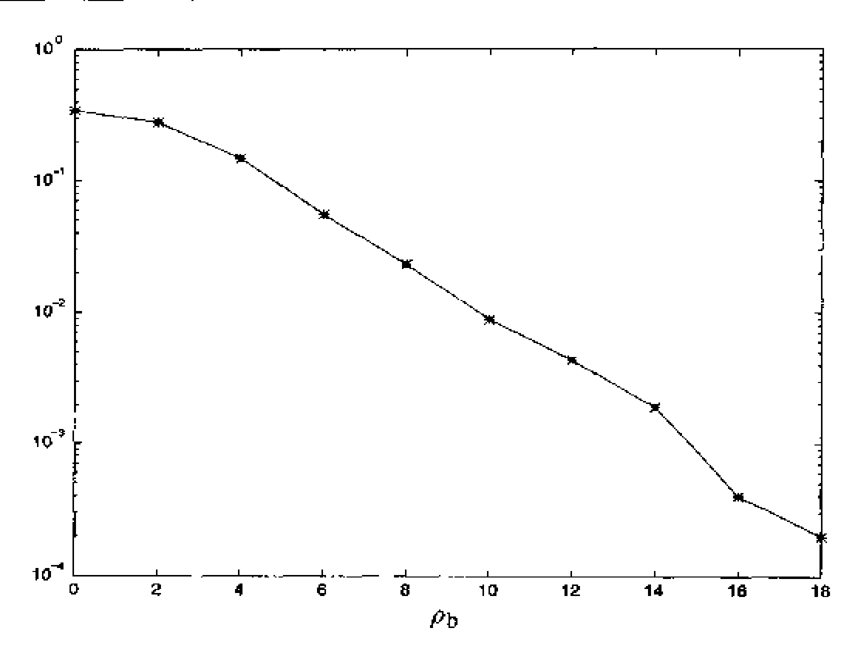


图 9.14　Monte Carlo 仿真:在部分频带干扰下,FH 双分集二进制 FSK 的误码率性能

m 文件

```
function [p]=ss_Pe97(rho_in_dB)
% [p]=ss_Pe97(rho_in_dB)
%              SS_PE97  finds the measured error rate. The value of
%              signal per interference ratio in dB is given as an input
%              to the function.
rho=10^(rho_in_dB/10);
Eb=rho;                                    % energy per information bit
E=Eb/2;                                    % energy per symbol transmitted
% the optimal value of alpha
if (rho>2),
  alpha=2/rho;
else
  alpha=1;
end;
% the variance of the additive noise
if (E>1),
  sgma=sqrt(E/2);
else
  sgma=sqrt(1/2);
end;
N=10000;                                   % number of bits transmitted
% generation of the data sequence
for i=1:N,
  temp=rand;
  if (temp<0.5)
    data(i)=1;
  else
    data(i)=0;
  end;
end;
% Find the transmitted signals.
for i=1:N,
  if (data(i)==0),
    tr11c(i)=sqrt(E);   tr12c(i)=sqrt(E);
```

```
    tr11s(i)=0;     tr12s(i)=0;
    tr21c(i)=0;     tr22c(i)=0;
    tr21s(i)=0;     tr22s(i)=0;
  else
    tr11c(i)=0;     tr12c(i)=0;
    tr11s(i)=0;     tr12s(i)=0;
    tr21c(i)=sqrt(E);          tr22c(i)=sqrt(E);
    tr21s(i)=0;     tr22s(i)=0;
  end;
end;

% Find the received signals, make the decisions, and count the number of errors made.
num_of_err=0;
for i=1:N,
  % determine if there is jamming
  if (rand<alpha),
    jamming1=1;                          % jamming present on the second transmission
  else
    jamming1=0;                          % jamming not present on the first transmission
  end;
  if (rand<alpha),
    jamming2=1;                          % jamming present on the second transmission
  else
    jamming2=0;                          % jamming not present on the second transmission
  end;
  % The received signals are
  if (jamming1==1)
    r11c=tr11c(i)+gngauss(sgma);      r11s=tr11s(i)+gngauss(sgma);
    r21c=tr21c(i)+gngauss(sgma);      r21s=tr21s(i)+gngauss(sgma);
  else
    r11c=tr11c(i);      r11s=tr11s(i);
    r21c=tr21c(i);      r21s=tr21s(i);
  end;
  if (jamming2==1)
    r12c=tr12c(i)+gngauss(sgma);      r12s=tr12s(i)+gngauss(sgma);
    r22c=tr22c(i)+gngauss(sgma);      r22s=tr22s(i)+gngauss(sgma);
  else
    r12c=tr12c(i);      r12s=tr12s(i);
    r22c=tr22c(i);      r22s=tr22s(i);
  end;
  % Compute the decision variables, first the weights.
  if (jamming1==1),
    w1=1/sgma^2;
  else
    w1=10;
  end;
  if (jamming2==1),
    w2=1/sgma^2;
  else
    w2=10;
  end;
  % The intermediate decision variables are computed as follows.
  r11=r11c^2+r11s^2;
  r12=r12c^2+r12s^2;
  r21=r21c^2+r21s^2;
  r22=r22c^2+r22s^2;
  % Finally, the resulting decision variables x and y are computed.
  x=w1*r11+w2*r12;
```

```
  y=w1*r21+w2*r22;
  % Make the decision.
  if (x>y),
    decis=0;
  else
    decis=1;
  end;
  % Increment the counter if this is an error.
  if (decis~=data(i)),
    num_of_err=num_of_err+1;
  end;
end;
% The measured bit error rate is then
p=num_of_err/N;
```

9.5 习题

9.1 写一个 MATLAB 程序,完成某 DS 扩频系统的 Monte Carlo 仿真,该系统用二进制 PSK 经由 AWGN 信道传输信息。假设处理增益是 10。画出测得的误码率和 SNR 的对比图,从而说明从该扩频信号中没有任何性能上的获益。

9.2 写一个 MATLAB 程序,完成某 DS 扩频系统的 Monte Carlo 仿真,该系统运行在 LPI 模式。处理增益是 20(13 dB),在信号解扩前接收端的期望功率信噪比 P_s/P_N 是 -5 dB 或更小。画出作为 SNR 的函数的已测误码率。

9.3 设处理增益为 10,重做解说题 9.4 的 Monte Carlo 仿真,并画出已测误码率。

9.4 写一个 MATLAB 程序,实现 $m=12$ 级的最大长度移位寄存器,并产生 3 个周期的序列。计算并画出由式(9.3.1)给出的等效双极性序列的周期自相关函数。

9.5 写一个 MATLAB 程序,实现 $m=3$ 级和 $m=4$ 级的最大长度移位寄存器,将它们的输出序列按模 2 相加。所得出的序列是周期的吗? 若是,序列的周期是什么? 利用式(9.3.1)计算并概略画出所得(双极性)序列的自相关序列。

9.6 写一个 MATLAB 程序,计算在解说题 9.5 中产生的 $L=31$ 的 Gold 序列的自相关序列。

9.7 写一个 MATLAB 程序,实现 4 个时间同步的 CDMA 用户的 Monte Carlo 仿真,其中每个用户都使用长度为 $L=31$ 的不同的 Gold 序列。这 4 个 Gold 序列可以从表 9.3 中挑选。每个用户都采用它们代表的 Gold 序列的二进制(± 1)调制。每个用户的接收端将这个复合的 CDMA 接收信号进行相关,而这个复合信号都在它们各自的序列上受到加性高斯白噪声的污损(基于按码片相加)。在 $N=10\ 000$ 信息比特下,用 Monte Carlo 仿真估值并画出作为 SNR 的函数的每个用户的差错概率。

9.8 当 4 个用户非同步传输时重做习题 9.7。例如,可以进行这样的仿真:4 用户 CDMA 信号在时间上相对偏移一个码片;也就是说,用户 2 的 CDMA 信号相对于用户 1 延迟了一个码片,用户 3 的 CDMA 信号相对于用户 2 的信号延迟了一个码片,用户 4 的 CDMA 信号相对于用户 3 的信号延迟了一个码片。试比较用非同步传输和同步传输所得到的差错概率。

9.9 一个 FH 二进制正交 FSK 系统采用 $m=7$ 级移位寄存器产生一个周期的长度为 $L=127$

的最大长度序列。该移位寄存器的每一级选择在跳频图上的 $N=127$ 个非重叠频带中的一个。写一个 MATLAB 程序，对中心频率的选取和在 $N=127$ 个频带的每一个中两个频率的产生进行仿真。给出前 10 个比特区间内的频率选择图。

9.10　写一个 Monte Carlo 程序，仿真某一 FH 数字通信系统，该系统采用二进制 FSK 并用非相干（平方律）检测。该系统遭受功率谱密度为 $J_0/\alpha(\alpha=0.1)$ 的部分频带干扰的侵扰。在该干扰频带 $0<\alpha\leqslant0.1$ 内，功率谱是平坦的。画出该系统测出的误码率与 SNR($\mathscr{E}_b/J_0$) 的对比图。

9.11　在解说题 9.7 中，当没有干扰时在相加器上所用的权系数是 $w=100$，而在有干扰时该权系数是 $w=1/\sigma^2=2/\mathscr{E}$，其中信号能量是 $\mathscr{E}\geqslant4$，重做这种情况下的 Monte Carlo 仿真。对于这个双分集系统，画出由 Monte Carlo 仿真测出的误码率，并将其性能与在解说题 9.7 中所得的性能进行比较。

第 10 章　数字调制方法的 Simulink 仿真指南

10.1　概述

本章以前面有关章节中讨论的频率载波调制方法的若干精选题为对象，为软件仿真提供一种辅导性的仿真指南。所有程序都是用 Simulink 和 MATLAB 编写的。征得 MathWorks 公司的许可，这个软件包中有源自 MATLAB 工具箱和 Simulink 功能块的若干功能和模块，而这些在 MATLAB 和 Simulink 的标准学生版中是不包括的。这样，就能让学生在 Windows 和 Linux 环境下，用 MATLAB 和 Simulink 标准学生版运行这些辅导材料中的有关内容。MathWorks 不支持这些功能和模块。对于不是 Windows 和 Linux 的其他平台，除了标准的 MATLAB/Simulink 学生版以外，还需要 Singnal Processing Toolbox，Communications Toolbox，Communications Blockset 和 DSP Blockset。

这个指南的目的是用例子来进一步阐明前面从理论上提到的各种数字调制方法，并通过在某些参数下的观察和实验更好地理解它们。只要有需要，我们就会给出附加的理论，而有些理论则是进一步的扩充，目的是为所讨论的问题提供某种不同的观点。根据本章所具有的辅导性的特点，我们不仅仅是展现这些内容，而是构造了一些能够促使学生主动学习的问题，使学生能够检验他们对问题的理解程度。本章中的各节都是按包含问题的理论部分和利用 Simulink 程序描述实验的实践部分来组织的。

由于很多学生对 Simulink 不如 MATLAB 熟悉，所以我们首先在 10.2 节对 Simulink 进行了简单介绍。仿真指南的内容基于图 7.18 的传输框图，并涵盖了脉冲成形(10.4 节)，BPSK(10.5 节)，QPSK(10.6 节)，Offset-QPSK(10.7 节)和 16-QAM(10.9 节)。10.8 节描述了一种二元正交频移键控(最小频移键控)(MSK)的偏移 QPSK 表示，可以证明 MSK 也能看成是一种线性调制方法。软件包中还包括了通 - 断键控和 4-ASK 仿真的内容。为了启动仿真过程，进入存有仿真文件的目录并在 MATLAB 工作空间键入 *DigModTutorial* 即可，这时会打开如图 10.13 所示的主菜单，从该菜单即可选取待仿真的部分。

10.2　Simulink 简介

这一节利用 BPSK 传输作为例子对 Simulink 进行了简单介绍。但是，我们只局限在 Simulink 的原另及其在这个仿真辅导中所要求的特点上，至于更为详尽的内容建议读者参考 Simulink 手册。这里给出的 Windows 界面的图保持了原样，Linux 界面的图则稍有不同。

Simulink 是一个提供图形接口的动态系统的仿真程序，在 MATLAB 环境下它是一种工具箱。动态系统通常是用方框图表示的，在图形界面中我们可以直接利用这样的方框图进行 Simulink 编程。在现在的情况下，这个动态系统就是在时变数据下的数字调制框图。

从 MATLAB 工作空间键入命令：

```
simulink
```

启动 Simulink，然后会打开 *Block Library Browser* 并提供 Simulink 库中的可用模块(见图 10.1)。

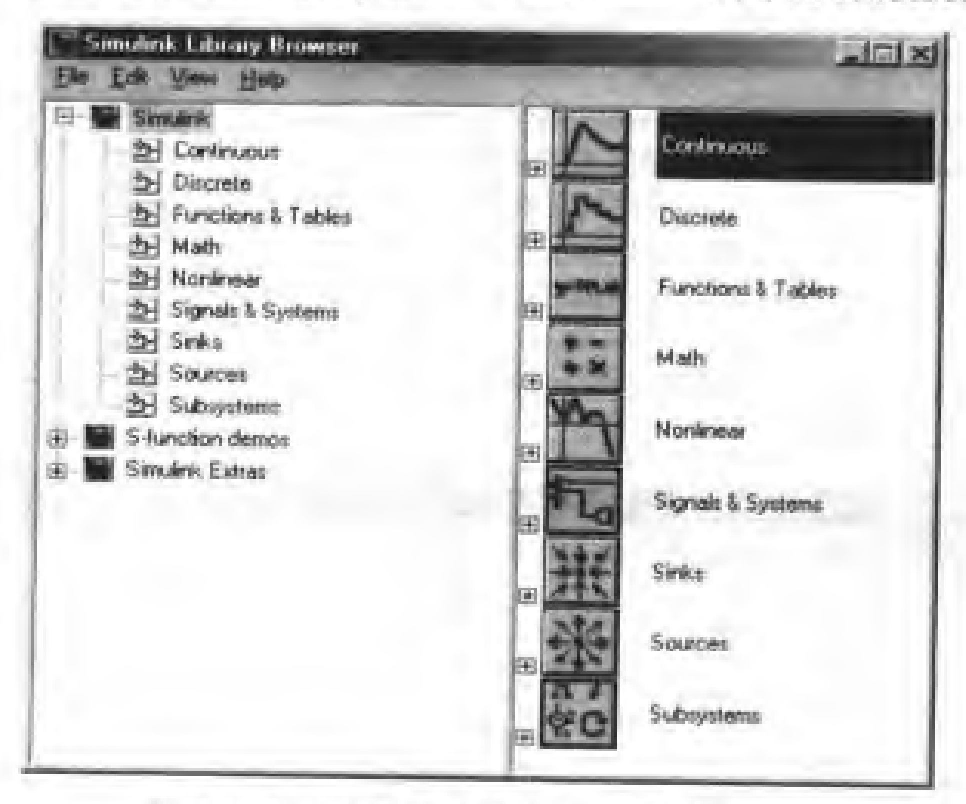


图 10.1 Simulink Block Library Browser(模块库浏览器)

这个库按模块的功能划分，例如 *Sources* 或 *Sinks*。在各自的框内双击就能打开某一个库。

10.2.1 举例：BPSK 传输

现在用采用矩形脉冲的 BPSK 传输来说明如何应用 Simulink，其目的是要构造如图 10.2 所示的 Simulink 模型。

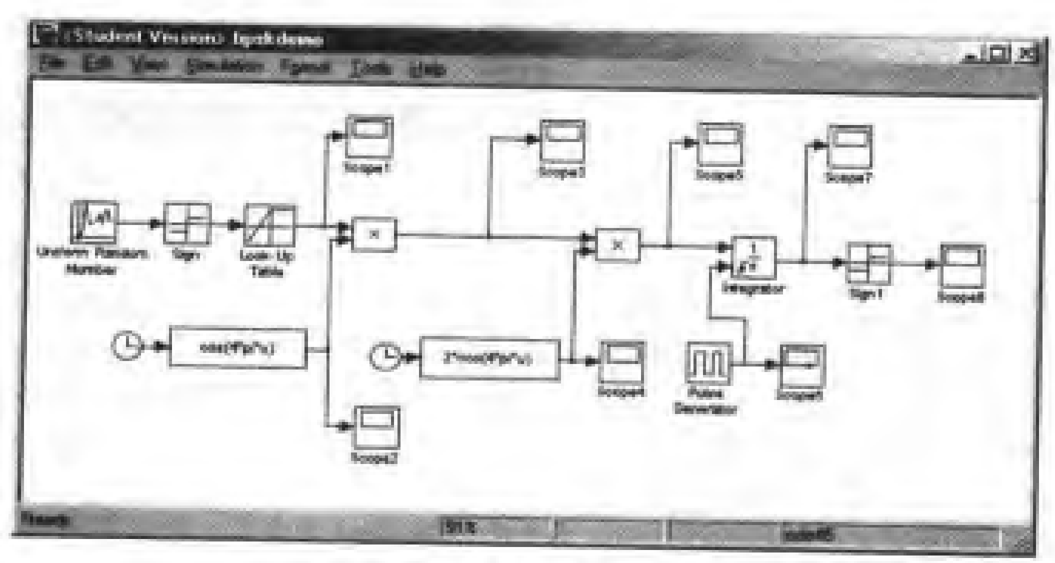


图 10.2 BPSK 传输的 Simulink 模型

为了创建一个新的 Simulink 模型,可以从菜单中打开一个新的模型:

File > New > Model

某个库模块可以从库窗口简单地拖放到 Simulink 模型窗口中。

利用 *Uniform Random Number Generator* 模块以及随后的 *Sign* 模块,可用做图 10.2 中的二进制信号源。*Uniform Random Number* 模块在指定的区间上产生均匀分布的随机数,在本例中该随机数在区间[-1,+1]内产生。跟在其后面的 *Sign* 模块的输出是:当输入大于 0 时输出为 1;若输入为 0 则输出为 0;当输入小于 0 时输出为 -1。为了产生输出比特为等概率的 +1 和 -1 二进制信号源,可以加入 *Look-Up Table* 模块,它的输入-输出映射关系为:+1 到 -1,-1 到 -1,而 0 到 +1。为了将模块 *Uniform Random Number* 加到模型中,可以在 Simulink Block Library Browser 中双击模块 *Sources*,这样就会显示出 *Sources* 库(见图 10.3)。将模块 *Uniform Random Number* 拖入 Simulink 模型并双击它。

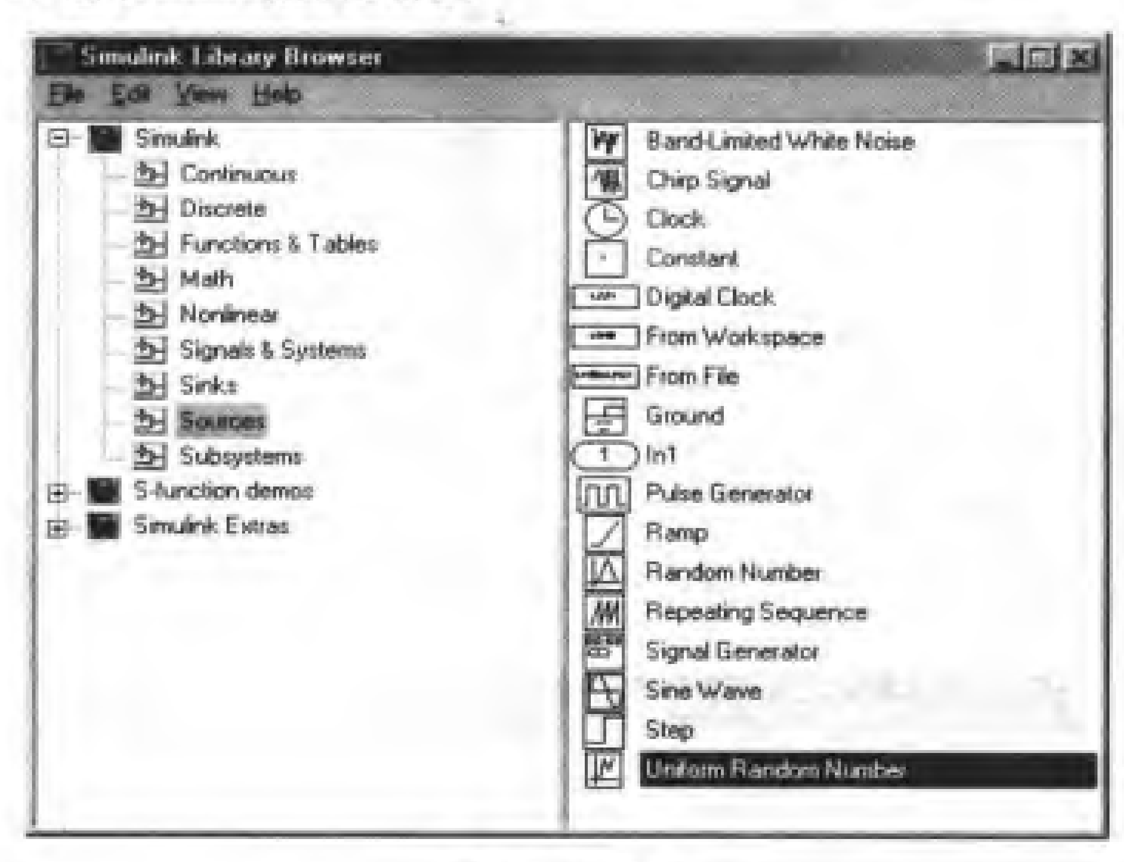


图 10.3　Sources 库

打开的 *Block Parameters* 窗口如图 10.4 所示,在这个窗口中即可输入如该图所示的有关参数。现将 *Math* 库中的 *Sign* 模块拖入模型。利用鼠标将模块 *Uniform Random Number* 和 *Sign* 连接起来:在模块 *Uniform Random Number* 的输出上按下左鼠标键并拖到 *Sign* 模块的输入端,然后释放鼠标键。*Look-Up Table* 模块从 *Functions & Tables* 库中得到,在模型中双击 *Look-Up Table* 模块并输入如图 10.5 所示的模块参数。

系统中任意一点的信号都能利用 *Sinks* 库中的模块 *Scope* 观察到。将一个 scope(相当于一台虚拟示波器)连接到 *Look-Up Table* 模块的输出端。

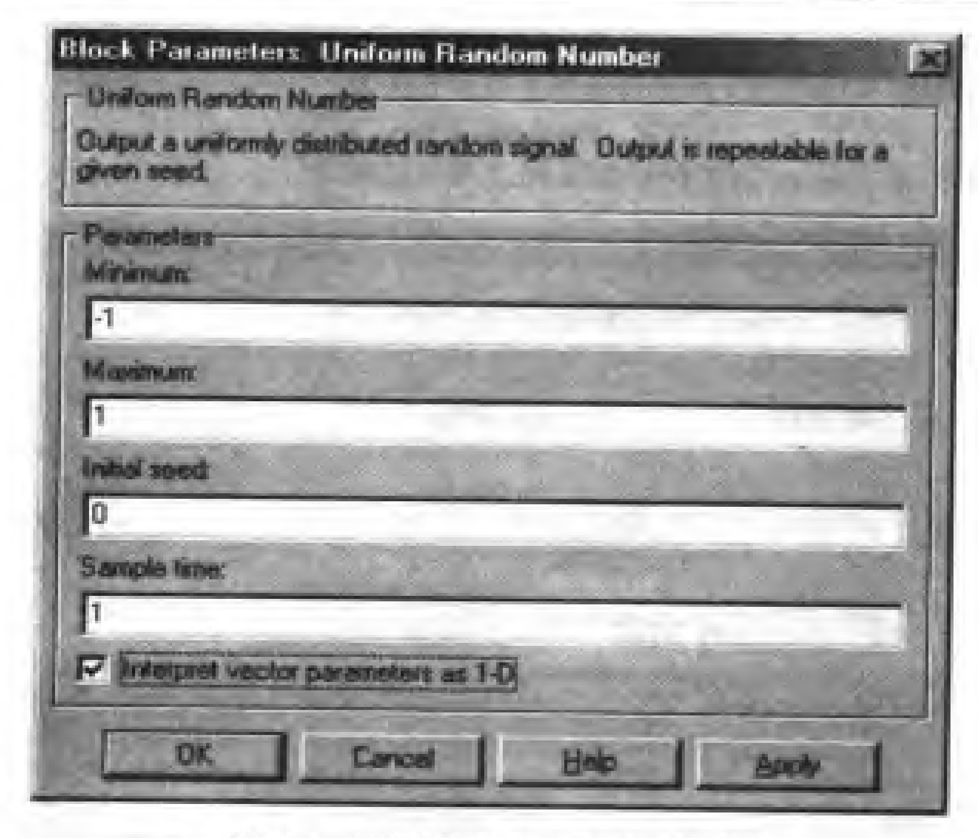


图 10.4　模块参数：Uniform Random Number

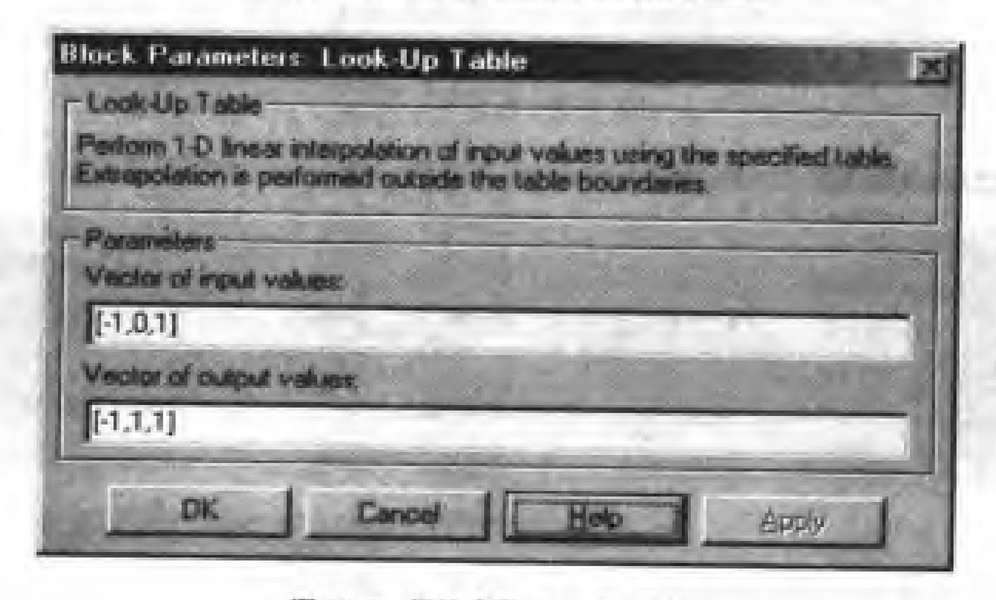


图 10.5　模块参数：Look-Up Table

在开始仿真之前，为了观察源信号，必须调整 *Simulation Parameters*。从菜单栏中使用命令：

Simulation > Simulation parameters

即可打开如图 10.6 所示的 *Simulation Parameters* 窗口。就目前而言，仅涉及参数 *Start time* 和 *End time*，稍后为了求得准确的结果，可能需要调整用于数值积分的 *Max step size* 和 *Min step size* 参数。从菜单栏中使用命令：

Simulation > Start

启动仿真。类似地,可以使用下面的命令结束或停止仿真:

Simulation > Stop

为了观察源信号,双击 scope 即可打开如图 10.7 所示的窗口。

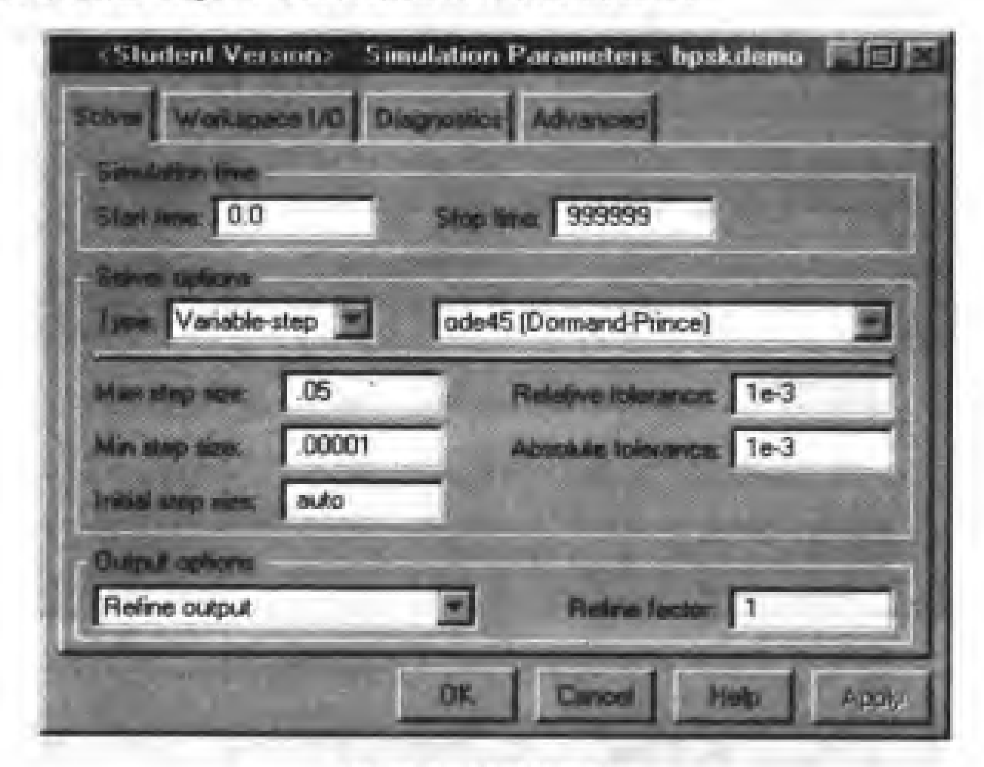


图 10.6 仿真参数窗口

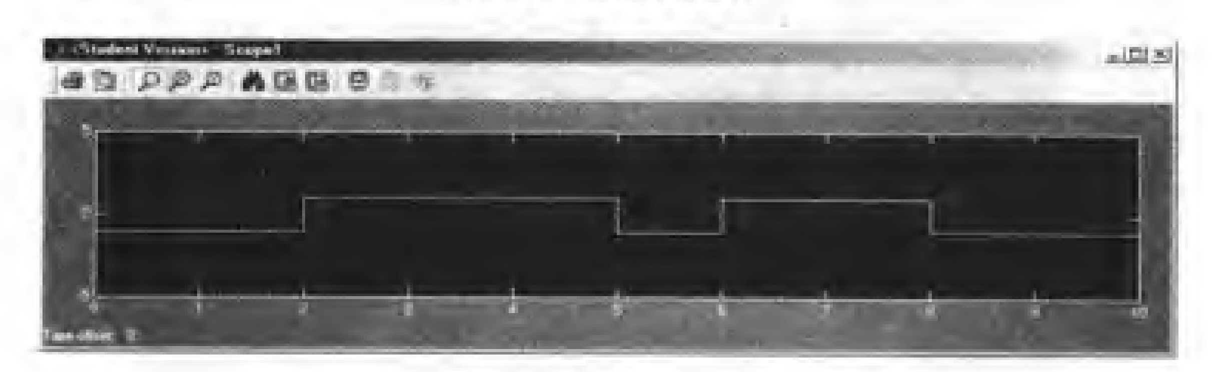


图 10.7 源信号(Scope1)

在窗口上单击右鼠标键会打开一个弹出菜单,从该菜单可以打开 *Axes properties* 窗口,如图 10.8所示,这是为了设定 y 轴的范围。

在图 10.7 中,*Scope* 窗口的时间范围可以通过菜单栏中的 *Parameters* 框来改变(见图 10.9)。

图 10.2 中的其他模块也能用相同的方式包括在内。利用 *Functions & Tables* 库中的 *Fcn* 模块可以产生载波 cos(4 * *pi* * *u*),该模块将某个指定的表达式(如 cos(*u*))加到输入变量 *u* 上。将模块 *Fcn* 拖入仿真模型并双击它,会立即打开 *Block Parameters* 窗口,如图 10.10 所示,在这里可以输入表达式 cos(4 * *pi* * *u*)。

<Student Version> : 'Scope1' pro...
Y-min: -5　Y-max: 5
Title ('%<SignalLabel>' replaced by signal name):
%<SignalLabel>
OK　Cancel　Apply

图 10.8　Axes properties 窗口(Scope1)

<Student Version> : 'Scope1' parameters
General　Data history　Tip: try right clicking on axes
Axes
Number of axes: 1　floating scope
Time range: 10
Tick labels: bottom axis only
Sampling
Decimation　1
OK　Cancel　Help　Apply

图 10.9　Scope1 的参数窗

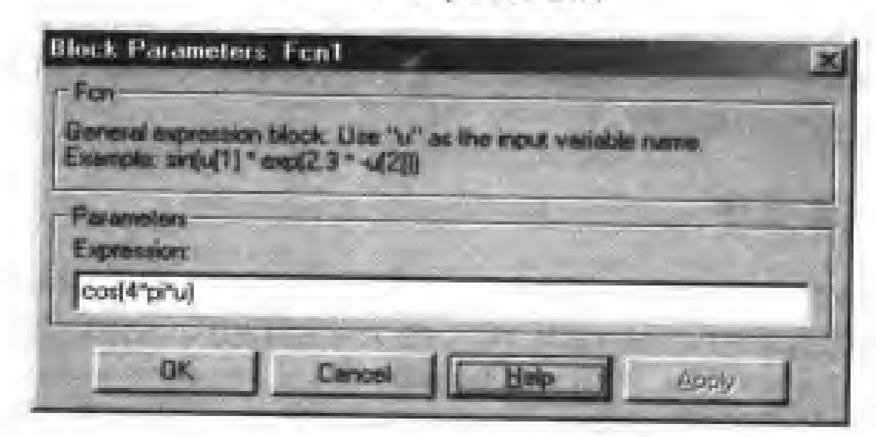


图 10.10　模块参数:Fcn

时间变量 u,可以从 *Sources* 库中的 *Clock* 模块得到。*Maths* 库中的 *Product* 模块用来产生已调信号。由于考虑的是无噪声传输仿真,因此就将发送端和接收端直接相连。如果有 *Communications Blockset* 可以利用[①],那么在发送端和接收端之间可以包含一个 AWGN 信道。接

① *Communications Blockset* 提供了通信系统仿真所需的一整套模块,而这些模块通常不包含在标准的 Simulink 版本中。

收端的匹配滤波器是利用 *Continuous* 库中的 *Integrator* 模块按积分清除来实现的。复位信号由 *Sources* 库中的 *Pulse Generator* 模块产生，其参数如图 10.11 所示。

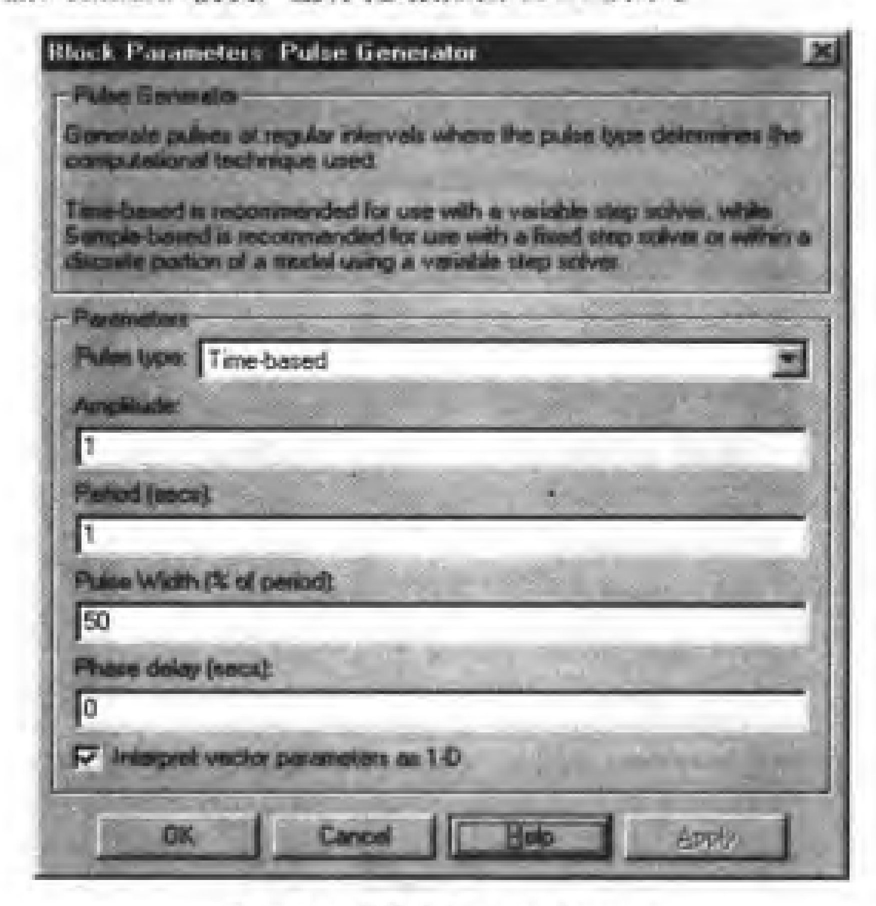


图 10.11　模块参数：Pulse Generators

右键单击一个模块，即可通过打开的弹出菜单获得对该模块的在线帮助。

现在可以从菜单栏开始仿真并观察不同传输点的示波器信号。为了确保信号准确，可能需要如图 10.6 所示改变仿真参数。

10.3　启动仿真程序

为了运行仿真程序，要将保存文件的目录以及所有子目录添加到 MATLAB 路径中。完成这个任务的最容易的方法是利用 *pathtool*，从 MATLAB 工作空间中用下面的命令即可将其打开：

pathtool

如图 10.12 所示，单击 *Add with Subfolders*…框并选取存有仿真文件的目录。该软件包中包含有源自 MATLAB 工具箱和 Simulink 模块的若干功能和模块，这样就允许用标准的 MATLAB 和 Simulink 学生版运行这个仿真程序。MathWorks 不支持这些功能和模块。为了避免混淆，建议从 MATLAB 路径中除掉与该仿真无关的所有 MATLAB 和 Simulink 工具箱的目录。

为了启动仿真程序，在 MATLAB 工作空间内键入：

DigModTutorial

这样就会打开如图 10.13 所示的主菜单，从该菜单内即可选择所要仿真的各个部分。

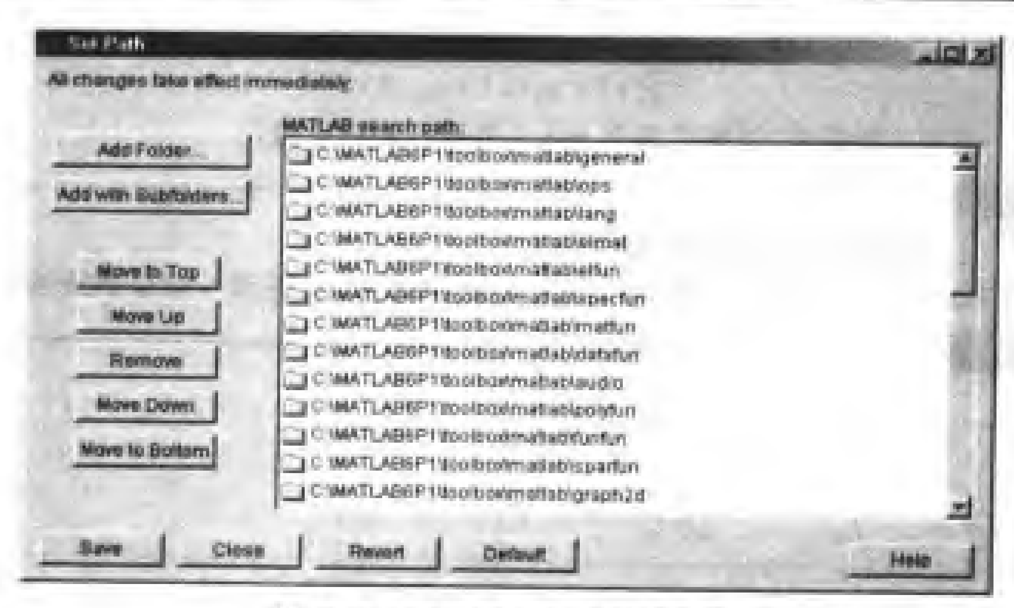


图 10.12　MATLAB pathtool(路径设置)

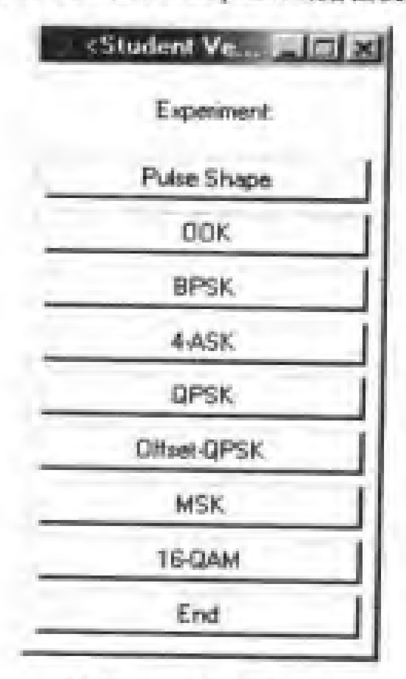


图 10.13　仿真主菜单

10.4　脉冲成形

这个仿真程序涵盖有不同脉冲成形的各种数字调制方法。第一部分给出了有关的脉冲及其傅里叶频谱。这一部分的目的是阐明无 ISI(码间干扰)传输与脉冲成形的关系，这是下面的实验的基础。这部分是用 MATLAB 写成的。

10.4.1　理论

发送滤波器的冲激响应 $g_T(t)$决定了发送信号的频谱特性。在时域中有限的冲激在频域中是无限的；在频域中是带限的脉冲在时域中是无限的。因此，为了在采样之后避免码间干

扰,发送和接收滤波器的级联:

$$g(t) = g_T(t) * g_R(t) \tag{10.4.1}$$

需要满足第一奈奎斯特条件,这些内容在 6.5.1 节中已经讨论过。

辅导题

辅导题 10.1　[奈奎斯特条件(零 ISI 条件)]

分别对冲激响应 $g(t)$和它的傅里叶变换 $G(f)$陈述零 ISI 的第一奈奎斯特条件。

题　解

$$g(nT) = \begin{cases} 1, & n = 0 \\ 0, & n \neq 0 \end{cases} \tag{10.4.2}$$

$$\sum_{m=-\infty}^{\infty} G\left(f - \frac{m}{T}\right) = T = \text{常数} \tag{10.4.3}$$

在这个仿真程序中用了两类满足零 ISI 条件的脉冲:NRZ(不归零)矩形脉冲和升余弦脉冲(见 6.5.1 节)。

辅导题

辅导题 10.2　[眼图]

在 6.4 节中引入眼图是作为观察 ISI 量的一种方法。试对二进制数据构造滚降系数 $\alpha = 0$ 的升余弦眼图。仅考虑两个相邻的干扰脉冲,哪些比特序列必须考虑?眼图中如何表现零 ISI?

题　解

升余弦脉冲(见图 10.14)为:

$$g(t) = \frac{\sin(\pi t/T)}{\pi t/T} \cdot \frac{\cos(\alpha \pi t/T)}{1 - 4\alpha^2 (t/T)^2}$$

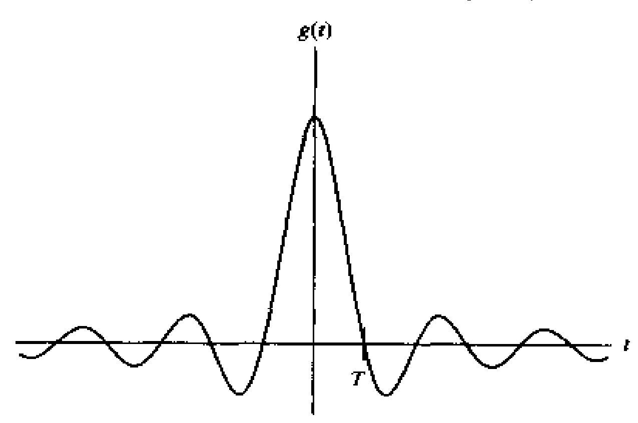


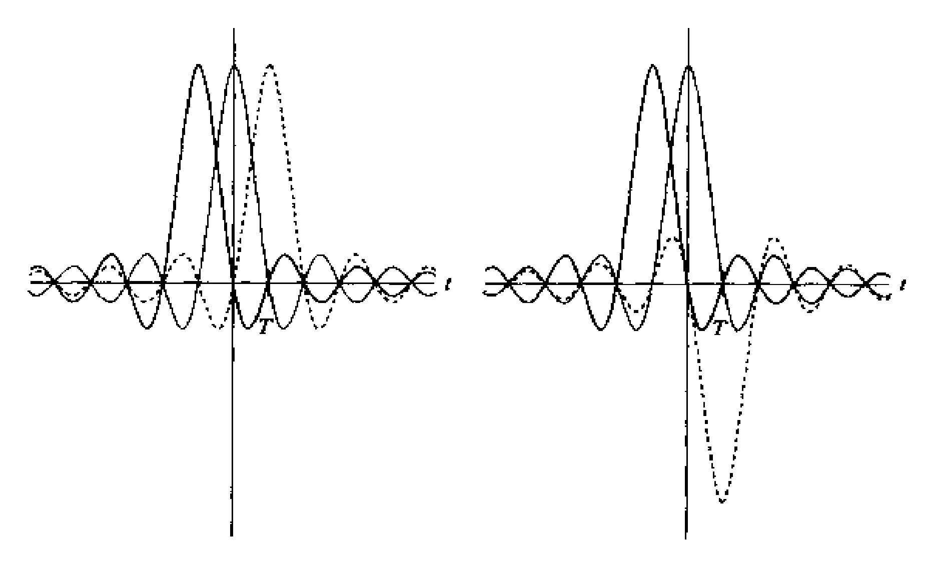
图 10.14　升余弦脉冲

以下 8 个比特序列必须要考虑:

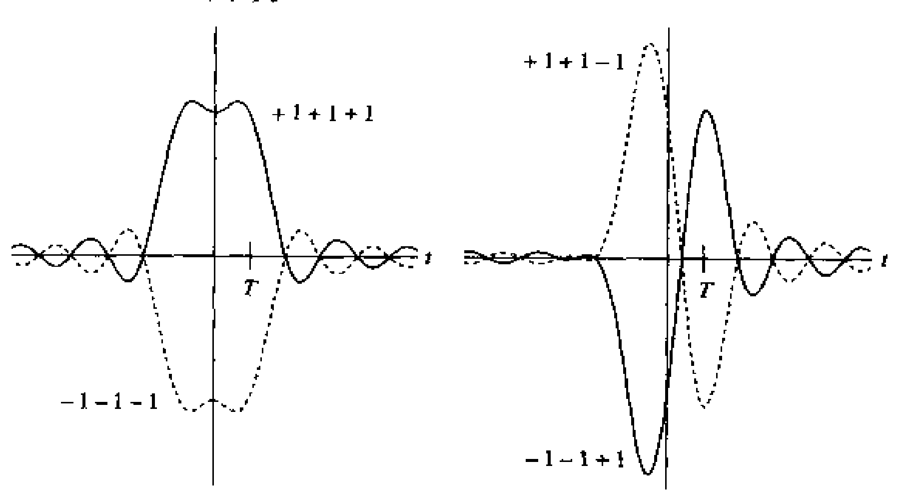
+1	+1	+1
+1	+1	−1
+1	−1	+1
+1	−1	−1

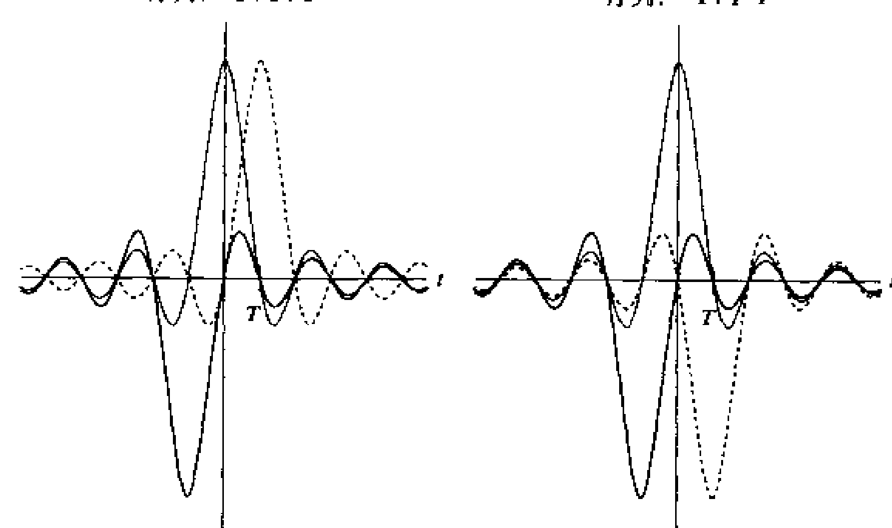
−1	+1	+1
−1	+1	−1
−1	−1	+1
−1	−1	−1

对 +1 位于中间比特位的 4 个比特序列，构造出的眼图（其余 4 个比特序列的贡献是对称的）为：

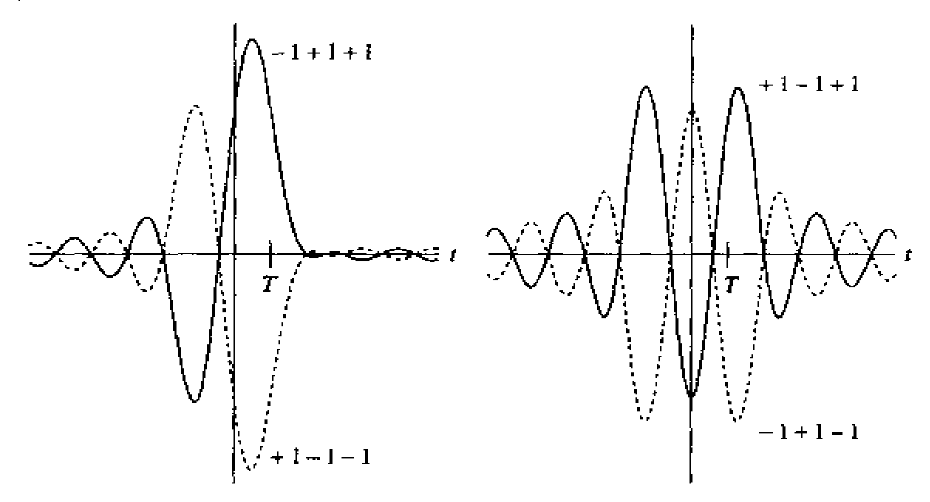


将脉冲相加得出的对眼图的贡献为：

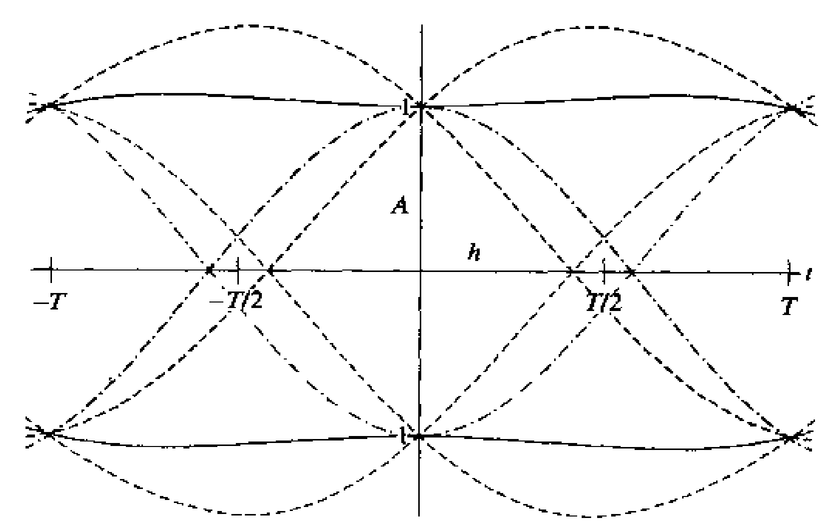




相加后的结果为：



眼图是全部比特序列贡献的叠加：



零 ISI 意味着最大的垂直眼图开启，也就是说，所有的线都在 +1 或 -1 点穿过。为了降低对定时误差的灵敏度，我们也希望有宽的水平眼图开启。例如，对 $\alpha=0.5$，水平眼图开启小于最大可能开启 T。

在 AWGN 信道中，使信噪比最大的最佳接收滤波器是匹配滤波器：

$$g_R(t)=g_T(-t) \tag{10.4.4}$$

因此，如果奈奎斯特脉冲成形在发送和接收滤波器之间是等量分割的，那么在接收滤波器输出端就得到了最大 SNR。于是傅里叶变换 $G_T(f)$应该是开根升余弦特性(见解说题 6.7)。

开根升余弦脉冲的冲激响应和傅里叶变换是：

$$g_T(t)=\frac{(4\alpha t/T)\cos[\pi(1+\alpha)t/T]+\sin[\pi(1-\alpha)t/T]}{(\pi t/T)[1-(4\alpha t/T)^2]} \tag{10.4.5}$$

$$G_T(f)=\begin{cases}\sqrt{T}, & 0\leqslant|f|\leqslant\dfrac{1-\alpha}{2T}\\ \sqrt{\dfrac{T}{2}\left[1+\cos\dfrac{\pi T}{\alpha}\left(|f|-\dfrac{1-\alpha}{2T}\right)\right]}, & \dfrac{1-\alpha}{2T}<|f|\leqslant\dfrac{1+\alpha}{2T}\\ 0, & |f|>\dfrac{1+\alpha}{2T}\end{cases} \tag{10.4.6}$$

如果开根升余弦发送滤波器用 $g_T(t)$，那么在接收滤波器之后存在零 ISI，但是发送信号会产生 ISI。

10.4.2　实验

为了启动仿真，先转到存有仿真文件的目录并在 MATLAB 工作空间中键入 *DigMod Tutorial*。这样会打开如图 10.13 所示的主菜单。从主菜单中选择 *Pulse Shape*，然后在 *Pulse Shape* 菜单中选中 *NRZ Rectangular*（见图 10.15）：

Pulse Shape > NRZ Rectangular

这样，一个 NRZ 矩形脉冲的脉冲形状和傅里叶变换就会展示出来，如图 10.16 所示。

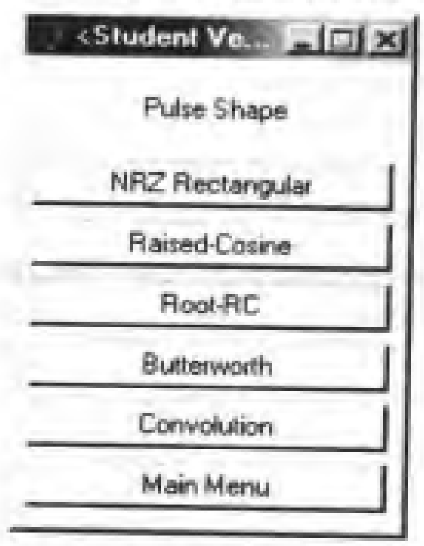


图 10.15　*Pulse Shape* 实验中的菜单

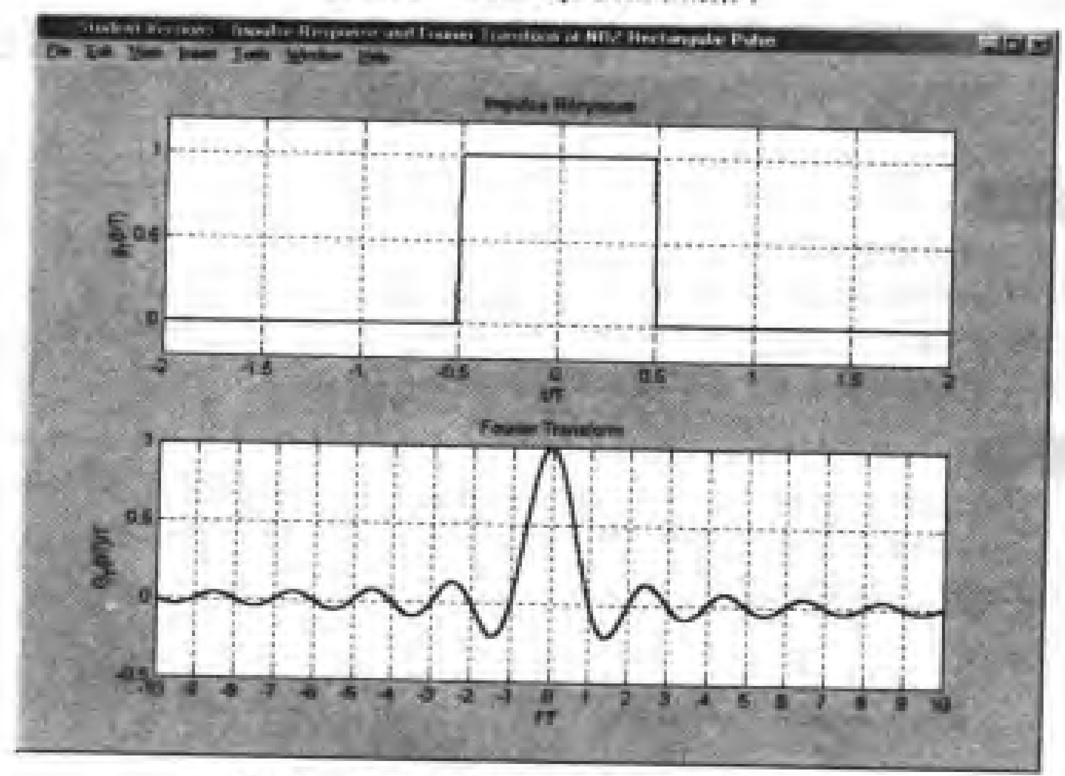


图 10.16　NRZ 矩形脉冲的冲激响应和傅里叶变换

辅导题

辅导题 10.3 [NRZ 矩形脉冲]

1. 满足零 ISI 条件吗?
2. NRZ 矩形脉冲是一个有限持续期的冲激响应,对带宽来说这意味着什么?

题 解

1. 满足零 ISI 条件,因为该脉冲限制到 T。
2. 有限持续期的冲激响应永远不会带限。

从 *Pulse Shape* 菜单中选取 *Raised-Cosine*

Pulse Shape > Raised-Cosine

观察在不同滚降系数 $0 \leqslant \alpha \leqslant 1$ 下的升余弦脉冲。利用滚动条可以改变滚降系数(见图 10.17)。包含在图中的绿色脉冲(虚线)是在时刻 T 发送的下一个比特的冲激响应。

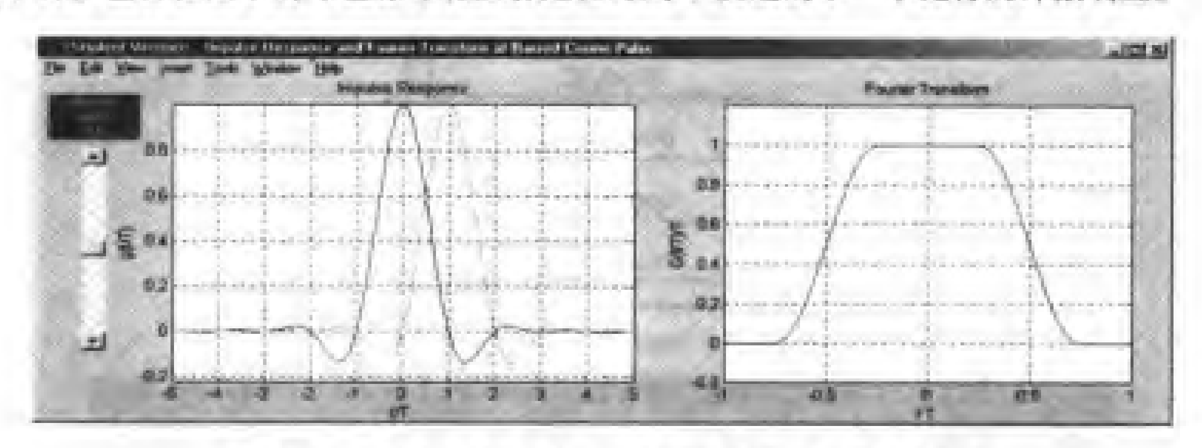


图 10.17 一种升余弦脉冲($\alpha = 0.5$)的冲激响应和傅里叶变换

辅导题

辅导题 10.4 [升余弦脉冲]

1. 对哪些滚降系数满足零 ISI 条件? 在冲激响应中怎样才能观察到这一点?
2. 哪些滚降系数产生最大的水平眼图开启? 在冲激响应中怎样才能看出这一点?
3. 滚降系数如何影响带宽? 在时域中影响什么?

题 解

1. 由于冲激响应在 nT, $n = 1,2,3,\cdots$,都是零,所以对全部 α 都满足零 ISI 条件。

2. 仅当 $\alpha = 1$ 时才有最大水平眼图开启 T。水平眼图开启在由 $+1 \to -1$ 和 $-1 \to +1$ 过渡时决定。由于当 $\alpha = 1$ 时冲激响应满足:

$$g\left(\frac{nT}{2}\right) = \begin{cases} 1, & n = 0 \\ \frac{1}{2}, & n = \pm 1 \\ 0, & \text{其余 } n \end{cases}$$

所以在眼图中各条线在 $T/2$ 时都为零。

3. 带宽随 α 增加而增大。在时域中冲激响应是无限长的，然而随 α 减小而快速下降。

现在检查具有各种滚降系数的开根升余弦脉冲(见图 10.18)。可以从 *Pulse Shape* 菜单上选取：


Pulse Shape > Root-RC

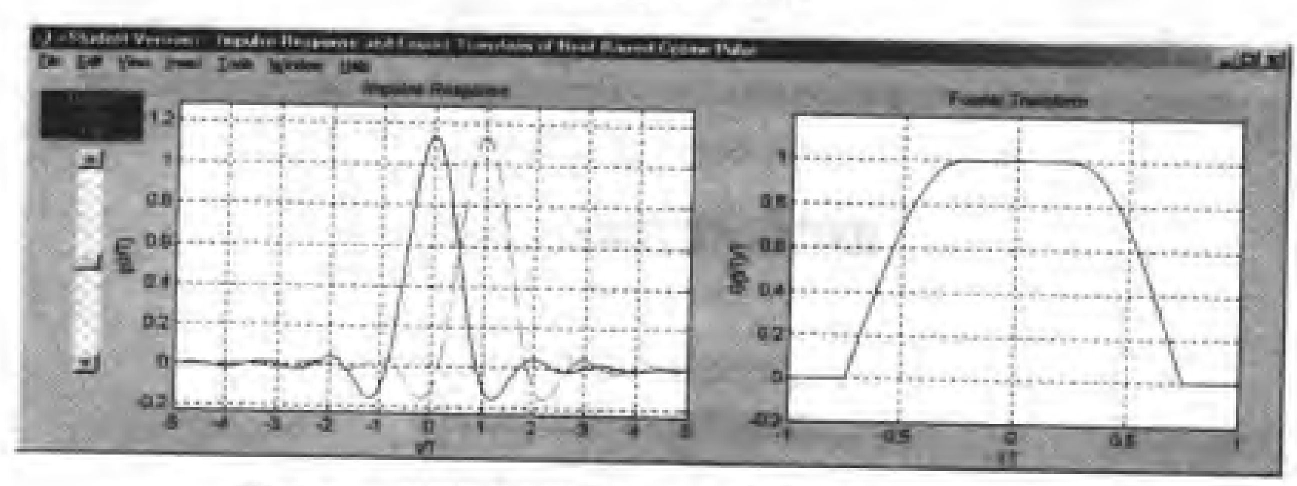


图 10.18　一种开根升余弦脉冲($\alpha=0.5$)的冲激响应和傅里叶变换

辅导题

辅导题 10.5　[开根 RC 脉冲]

1. 零 ISI 条件和最大水平眼图开启条件满足吗？
2. 对于哪个滚降系数，开根升余弦脉冲与升余弦脉冲是相同的？

题　解

1. 仅当 $\alpha=0$ 时满足零 ISI 条件，最大水平眼图开启总是小于 T。
2. $\alpha=0$。

在仿真中有时采用 4 阶巴特沃斯接收滤波器。通过键入：

Pulse Shape > Butterworth

即可展示出一个巴特沃斯滤波器的冲激响应和频率响应(见图 10.19)。

通过卷积可求得传输总冲激响应为：

$$g_T(t) * g_R(t) \tag{10.4.7}$$

这个总冲激响应可以利用 *Convolution* 菜单观察到(见图 10.20)：

Pulse Shape > Convolution > NRZ Rectangular * NRZ Rectangular

Pulse Shape > Convolution > Raised-Cosine * Raised-Cosine

Pulse Shape > Convolution > Root-RC * Root-RC

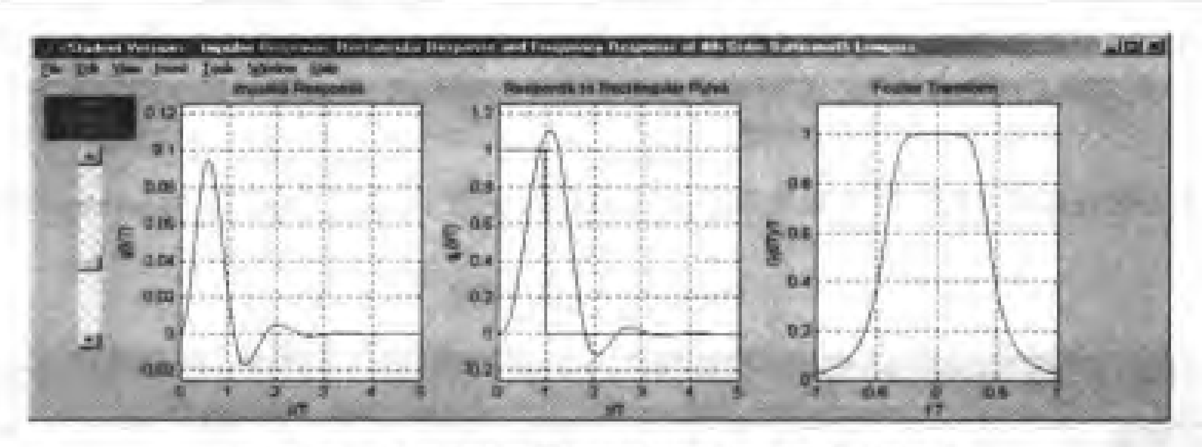


图 10.19 4 阶巴特沃斯低通波器的冲激响应和傅里叶变换

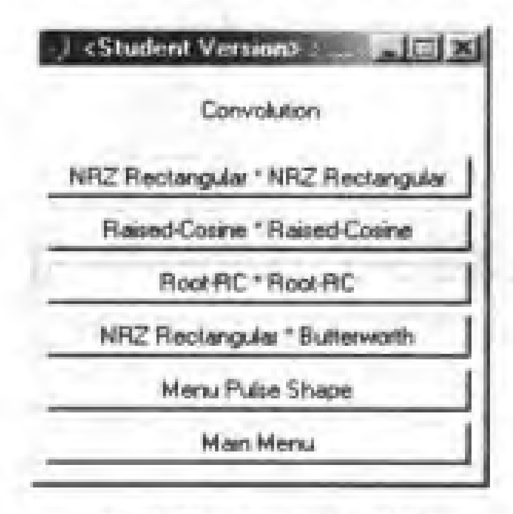


图 10.20 *Convolution* 菜单

辅导题

辅导题 10.6 [卷积]

1. 如果采用匹配滤波器,哪些脉冲形状满足零 ISI 条件?

2. 可以观察到,在匹配滤波器和小滚降系数 α 下,对于开根升余弦脉冲,零 ISI 条件不能真正满足,试进行解释!

题 解

1. NRZ Rectangular * NRZ Rectangular

   Root-RC * Root-RC,全部 α

   Raised-Cosine * Raised-Cosine, $\alpha=0$

2. 开根升余弦脉冲必须用有限长冲激响应实现,而在小滚降系数 α 时冲激响应在时域中衰减很慢,因此由于时域脉冲截尾而引起的误差不可忽略。

各种脉冲形状下的总冲激响应分别示于图 10.21(两个矩形脉冲)、图 10.22(两个升余弦脉冲)和图 10.23(两个开根升余弦脉冲)。

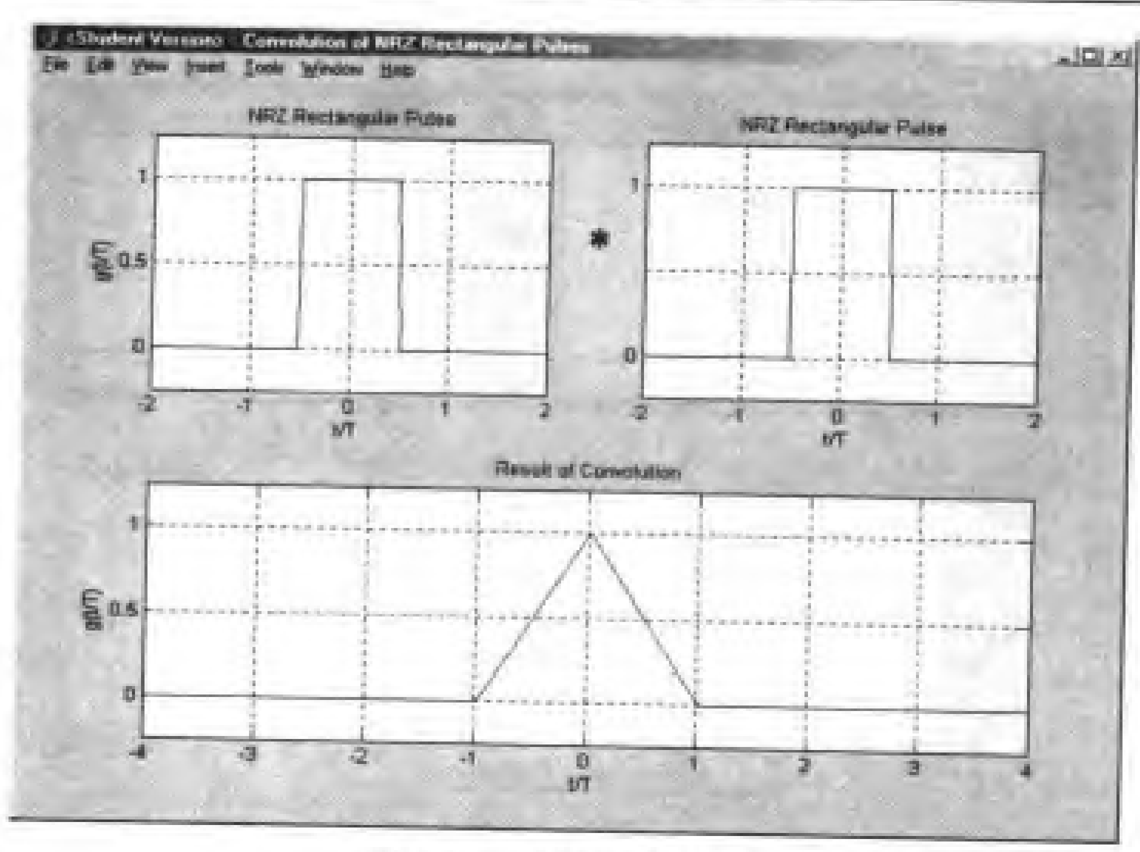


图 10.21　两个 NRZ 脉冲的卷积

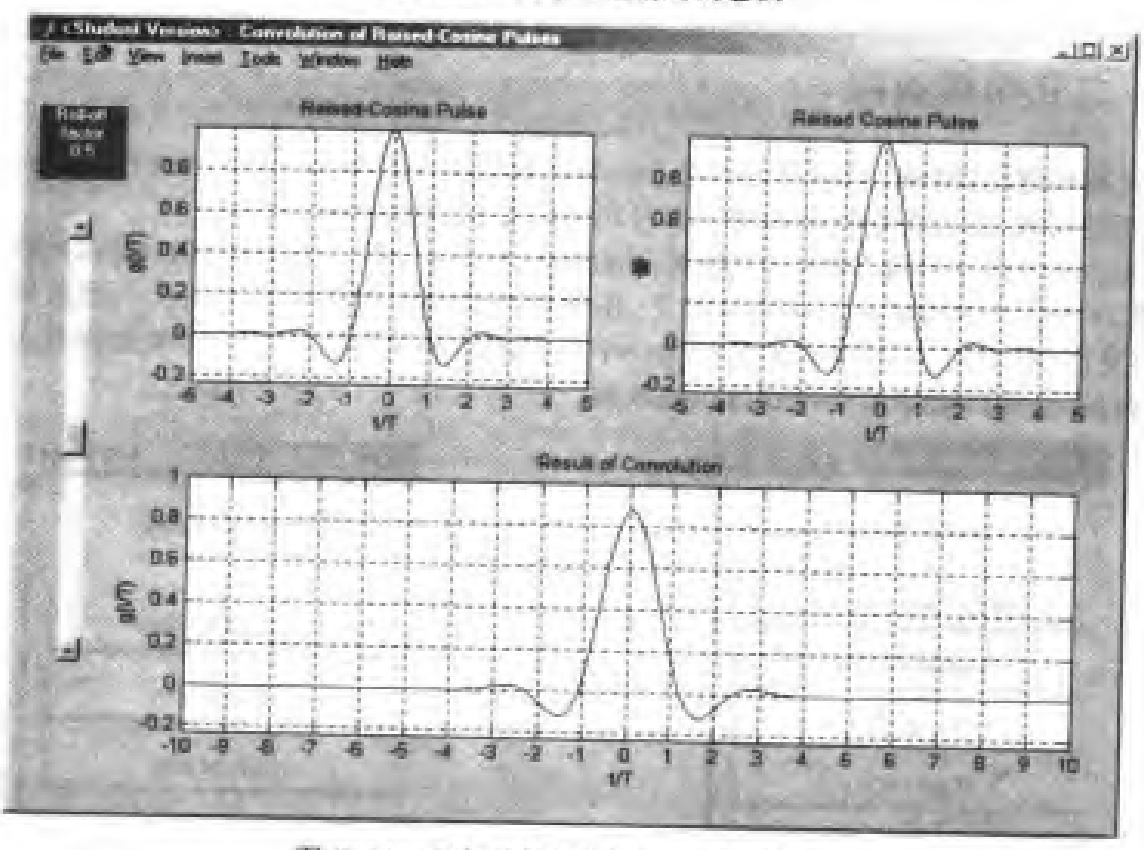


图 10.22　两个升余弦脉冲($\alpha=0.5$)的卷积

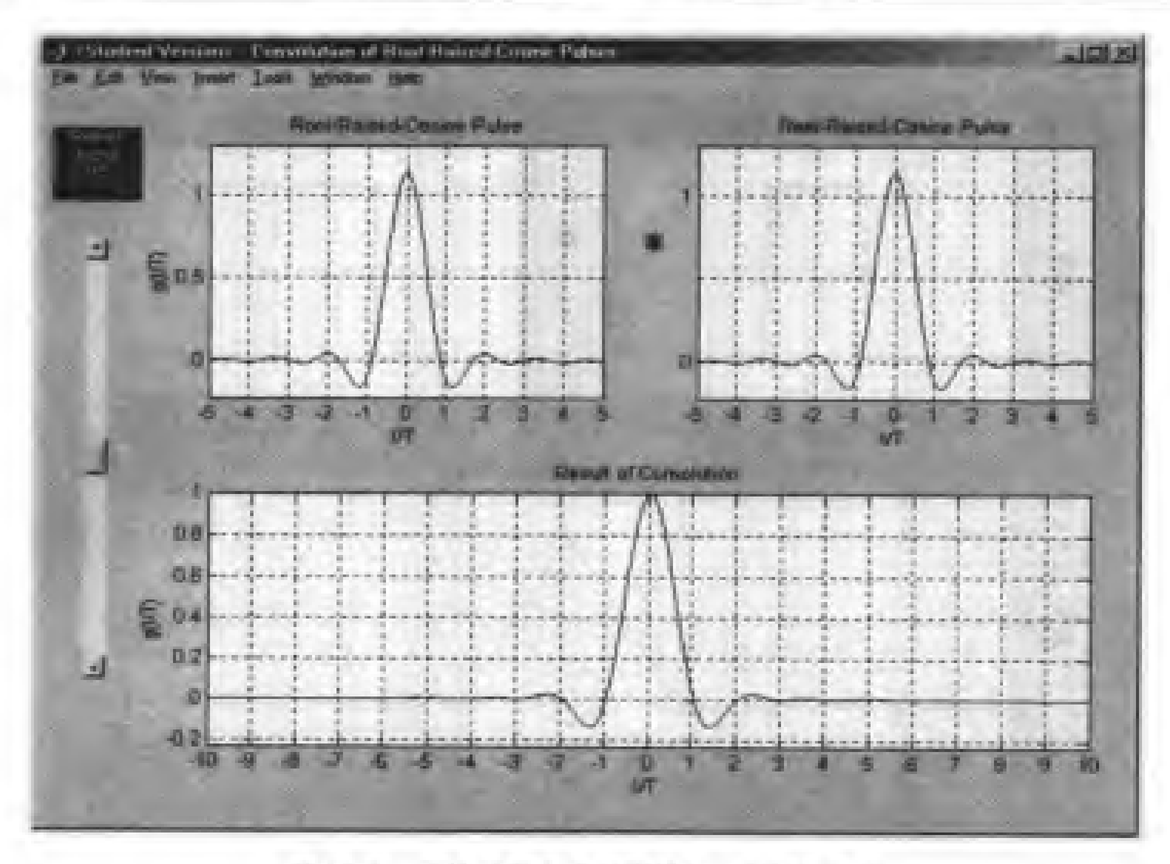


图 10.23 两个开根升余弦脉冲($\alpha=0.5$)的卷积

10.5 二元相移键控

二元相移键控(BPSK)可以作为熟悉 Simulink 软件的一种手段,在以后的各节中会再考虑很多方面的问题,并且我们将在本节的末尾对这些方面进行归纳。

首先从主菜单中选取 BPSK。这样将会显示出另一个菜单,从这个菜单中可以选择脉冲形状(见图 10.24)。在 NRZ 矩形脉冲情况下,还能在匹配滤波器和巴特沃斯接收滤波器之间进行选择(见图 10.25)。匹配滤波器是作为一个积分清除滤波器实现的。如果选择升余弦或开根升余弦脉冲,则匹配滤波器用做接收滤波器。

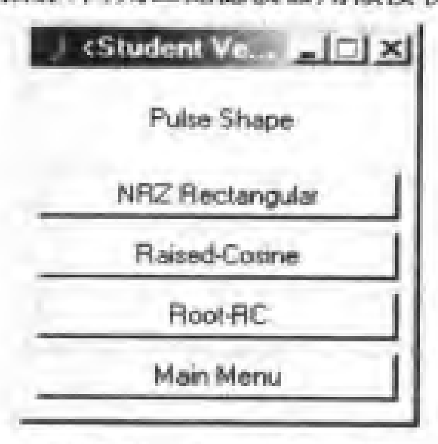


图 10.24 *Transmit filter* 菜单

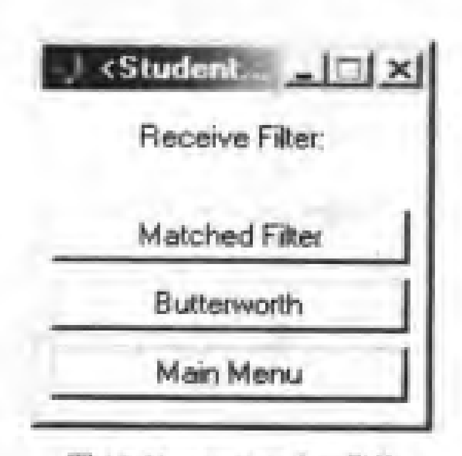


图 10.25 *Receive filter* 菜单

10.5.1 采用 NRZ 矩形脉冲的二元相移键控

选择 NRZ 矩形脉冲和匹配滤波器：

BPSK > NRZ Rectangular > Matched Filter

就会显示 Simulink 模型(见图 10.26)。在蓝色框 *Signal Space Constellation* 上双击即可显示 BPSK 星座图，从下拉菜单 *Simulation* 启动仿真：

Simulation > Start

发送端和接收端的眼图与信号空间图都将显示出来。各个窗口可以重叠，为了将另一个窗口放在前面可以用鼠标进行拖动。

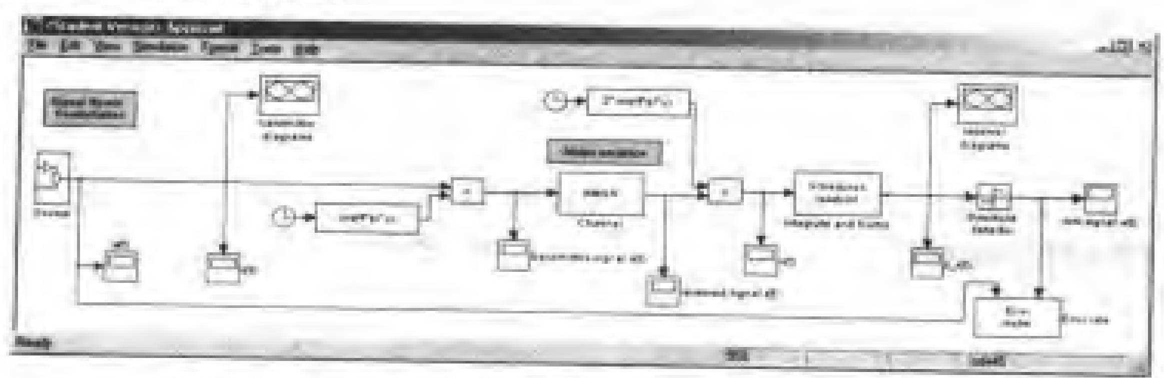


图 10.26 使用 NRZ 矩形脉冲和匹配滤波器的 BPSK 的 Simulink 模型

首先看一下时域信号，这可以通过在各自的 scope 上双击显示出来(见图 10.27 到图 10.29)。

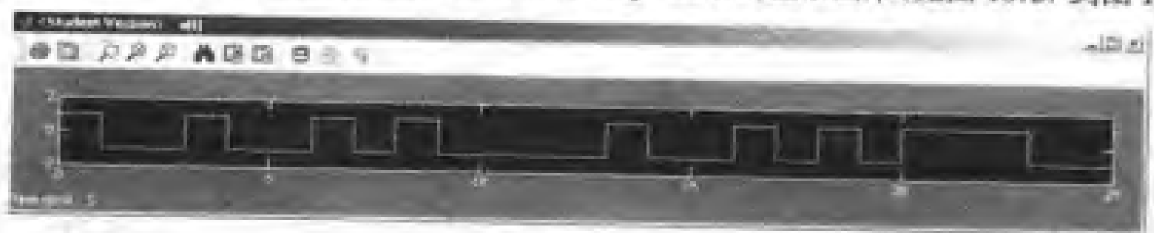


图 10.27 源信号 $a(t)$

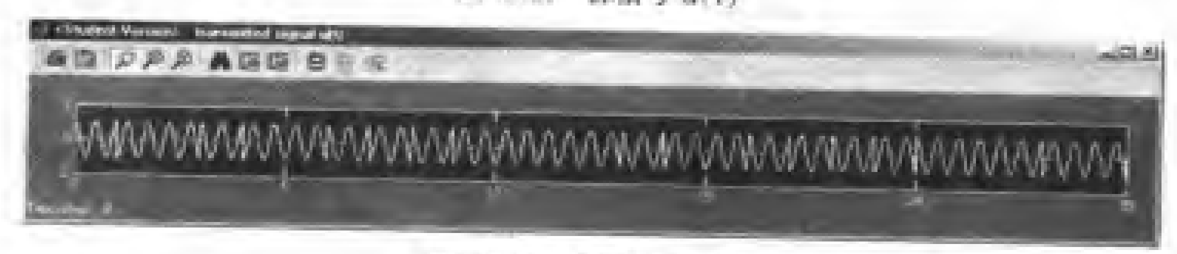


图 10.28 发送信号 $u(t)$

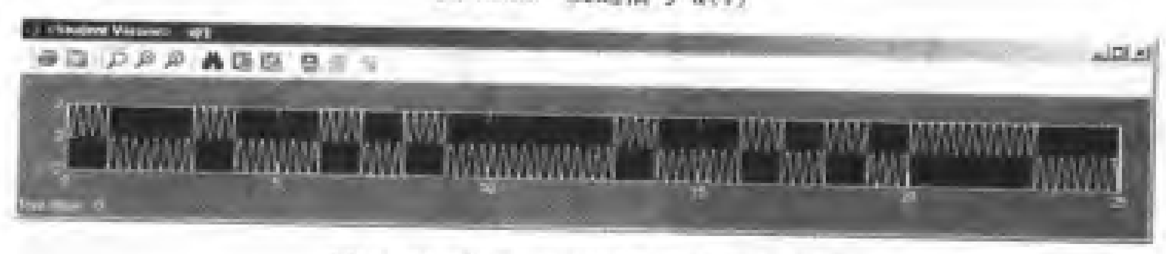


图 10.29 信号 $x(t)$(匹配滤波器输入端)

辅导题

辅导题 10.7 [采用 NRZ 矩形脉冲和匹配滤波器的 BPSK:时域信号]

1. 发送信号 $u(t)$的包络是常数吗?
2. 解释发送信号 $u(t)$和接收信号 $r(t)$不一致的原因。
3. 说明信号 $x(t)$的形式。

题 解

1. 包络是常数。
2. 由于接收信号受到加性噪声的污损(AWGN 信道)。
3. 经解调后信号 $x(t)$还包含有 $2f_c$ 的分量:

$$x(t) = s(t)\cos(2\pi f_c t)\cdot 2\cos(2\pi f_c t)$$
$$= s(t)(1+\cos(4\pi f_c t)) + \text{噪声}$$

现在来看看眼图和信号空间图(见图 10.30 至图 10.34)。

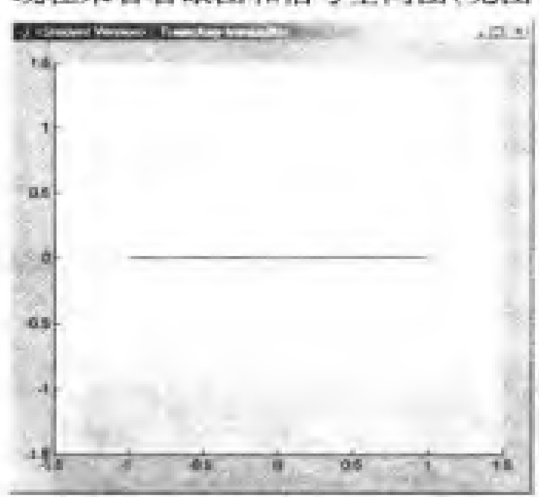


图 10.30 使用 NRZ 矩形脉冲的 BPSK 的发送端信号矢端轨迹

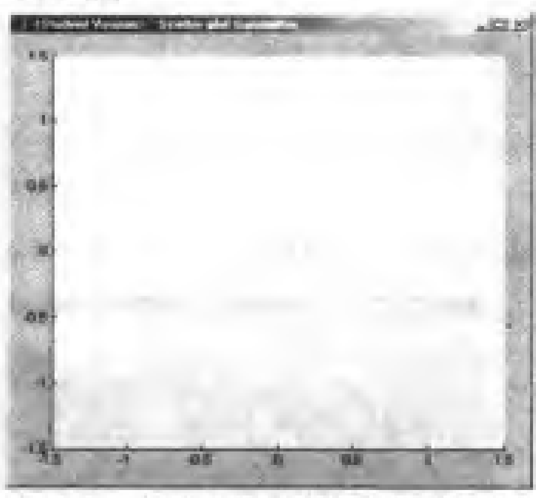


图 10.31 使用 NRZ 矩形脉冲的 BPSK 的发送端散布图(信号空间星座图)

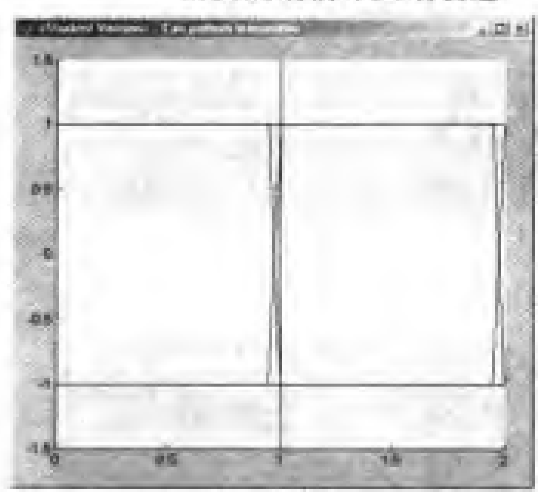


图 10.32 使用 ZRZ 矩形脉冲的 BPSK在发送端的眼图

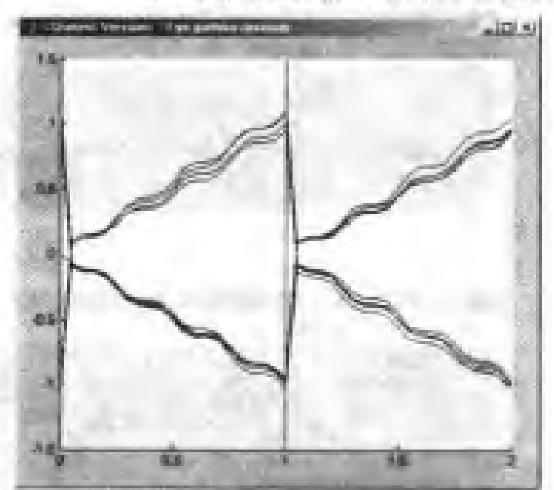


图 10.33 使用 NRZ 和匹配滤波器的BPSK在接收端的眼图

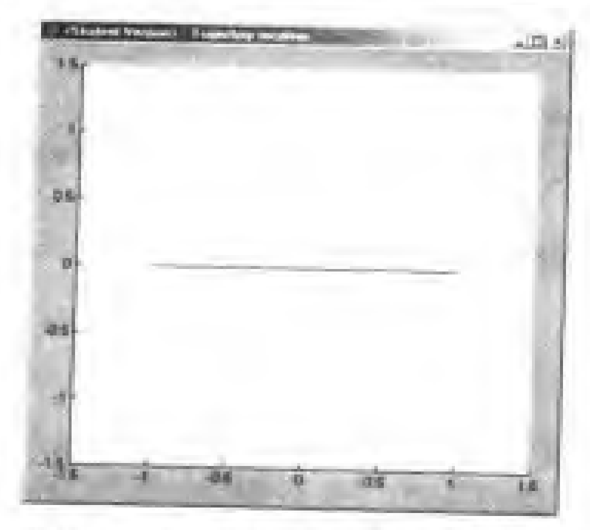


图 10.34 使用 NRZ 矩形脉冲和匹配滤波器的 BPSK 的接收端信号矢端轨迹

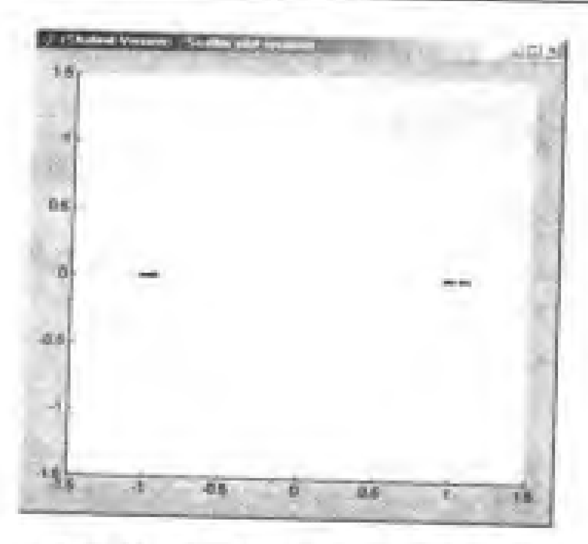


图 10.35 使用 NRZ 矩形脉冲和匹配滤波器的 BPSK 的接收端散布图

图 10.30 中的绿色矢端轨迹描述的是在信号空间中脉冲成形以后的基带信号。在最佳采样时间(即最大眼图开启处)用符号速率对绿色矢端轨迹采样，就得到了散布图中的蓝色信号空间点。就零 ISI 来说，能够精确地得到信号空间星座点。信号矢端轨迹描述了星座点之间的转移。

首先看一下发送端的图。

辅导题

辅导题 10.8 [采用 NRZ 矩形脉冲和匹配滤波器的 BPSK：发送端信号矢端轨迹、散布图和眼图]

1. 在发送端满足零 ISI 条件吗？在眼图和散布图中如何观察到这一点？
2. 水平眼图开启是最大的吗？

题 解

1. 满足，可得到最大垂直眼图开启度。在最佳采样时刻对信号矢端轨迹采样得到 BPSK 的星座点。
2. 水平眼图开启是最大的。

现在检查在接收端的各个图。

辅导题

辅导题 10.9 [采用 NRZ 矩形脉冲和匹配滤波器的 BPSK：接收端信号矢端轨迹、散布图和眼图]

1. 说明眼图的形式。
2. 水平和垂直眼图开启是最大的吗？
3. 最佳采样时刻是什么？

4. 为什么在接收端的散布图中观察到多于两个的信号空间点?

题　解

1. 接收滤波器是作为积分清除滤波器实现的,由 $x(t)$积分得到波形。另外,在接收信号中含有噪声。

2. 由于存在加性噪声,看不到最大水平和垂直眼图开启。

3. 当 $t/T=0,1,2,\cdots$时。

4. 由于加性噪声的缘故。

现在考虑无噪声传输。在 Simulink 模型中双击蓝色方框 *Noise variance* 可变化 AWGN 信道的噪声方差。通过显示的滚动条能将噪声电平调至最小可能值(零是不可能的,因为这样会产生奇异性)。从下拉菜单中选取下面的命令即可重新启动仿真:

Simulation > Stop

Simulation > Start

辅导题

辅导题 10.10　[采用 NRZ 矩形脉冲和匹配滤波器的无噪声 BPSK 传输]

1. 现在,发送信号 $u(t)$和接收信号 $r(t)$是如何相差的?

2. 在眼图和散布图上会看到哪些改变?

题　解

1. 现在,$u(t)$和 $r(t)$是一样的。

2. 发送端的各图没有任何改变。由于在接收端不存在噪声,所以眼图开启最大,且通过信号矢端轨迹的采样得出了 BPSK 信号星座点(由于积分的关系会稍微有点不准确)。

现在回到主菜单,选取巴特沃斯低通滤波器作为接收滤波器,取代了匹配滤波器:

BPSK > NRZ Rectangular > Butterworth

试比较这时源信号 $a(t)$和宿(sink)信号 $v(t)$。为此,可以直接将源信号窗放在宿信号窗上面。

辅导题

辅导题 10.11　[采用 NRZ 矩形脉冲和巴特沃斯接收滤波器的 BPSK]

1. 解释说明 $a(t)$和 $v(t)$之间的延迟。

2. 能看到检测差错吗?

3. 对于巴特沃斯接收滤波器,重做辅导题 10.9 和辅导题 10.10。

题　解

1. 巴特沃斯低通滤波器是一个部分响应滤波器,它在某种因果实现下会引入延迟。

2. 检测差错取决于噪声方差。

3. ● 眼图形状由使用巴特沃斯滤波器过滤决定,另外还有噪声存在。

　● 由于巴特沃斯滤波器,不再满足零 ISI 条件。

- 最佳采样时刻在最大垂直眼图开启处。
- 由于不满足零 ISI 条件，即使无噪声存在，接收信号空间点也将散布开。

现在知道了如何使用这个软件，因此下面各节将仅描述新内容，并指出应该注意什么。

10.5.2　采用升余弦脉冲的二元相移键控

这一节要研究采用升余弦脉冲的 BPSK。选择：

BPSK > Raised-Cosine

Simulink 模型如图 10.36 所示，滚降系数设定在 $\alpha=0.5$，滤波器延时是 6 个符号周期，即 $6T$，在全部仿真中都保持这种设定。然而，利用滚动条可以改变这两个设定值，双击蓝色方框 *Rolloff-Factor* 即可打开一个滚动条。

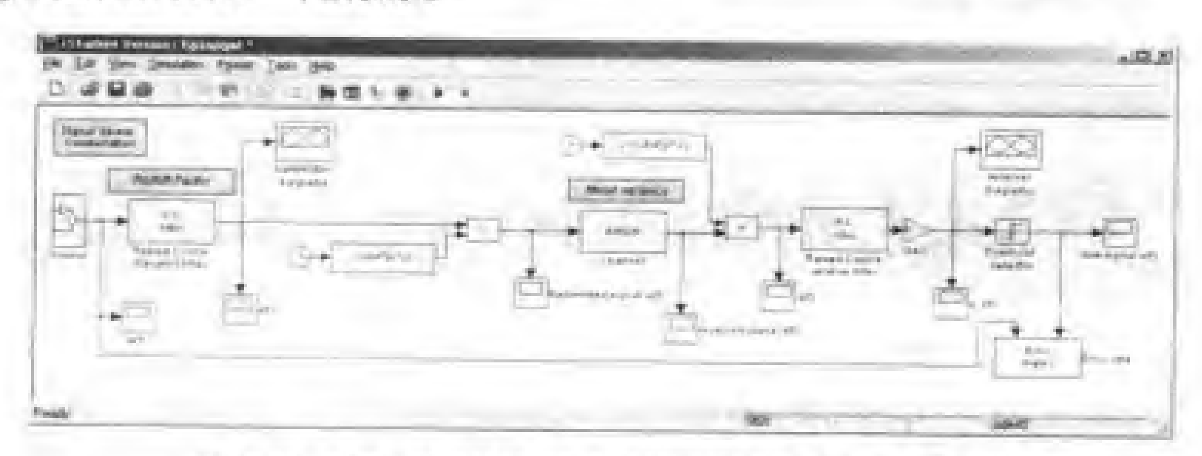


图 10.36　采用升余弦脉冲和匹配滤波器的 BPSK 的 Simulink 模型

将噪声方差置于很小的值。这样，时域信号、眼图和散布图如图 10.37 至图 10.46 所示。

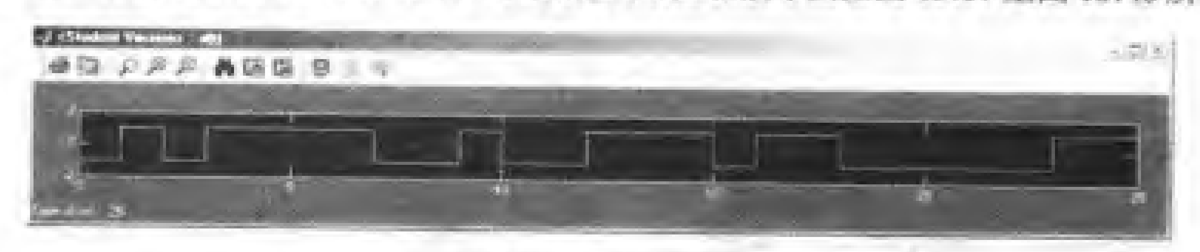


图 10.37　源信号 $a(t)$

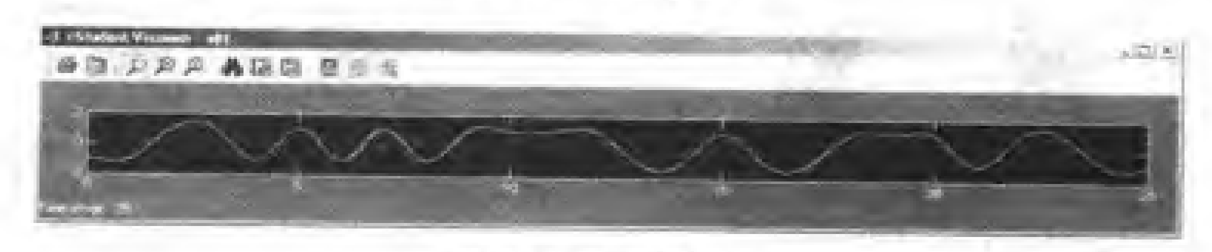


图 10.38　基带信号 $s(t)$

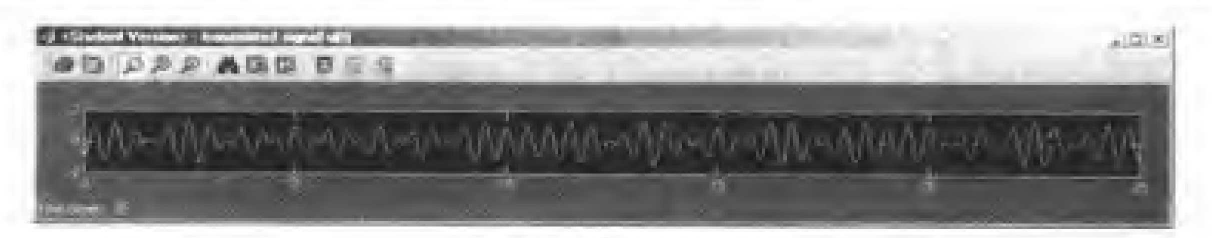


图 10.39 发送信号 $u(t)$

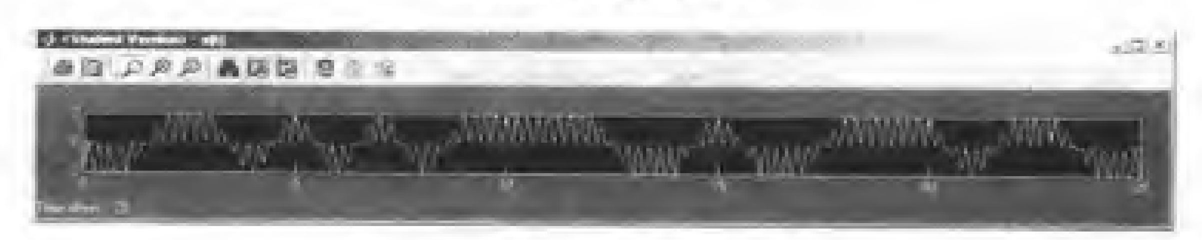


图 10.40 信号 $x(t)$(匹配滤波器输入端)



图 10.41 已解调基带信号 $x_r(t)$(匹配滤波器输出端)

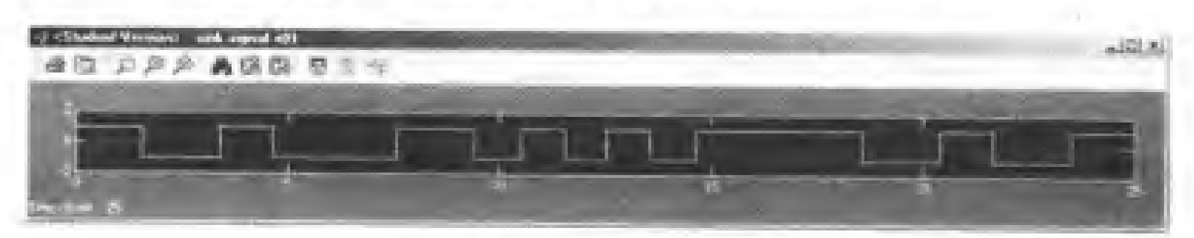


图 10.42 宿信号 $v(t)$

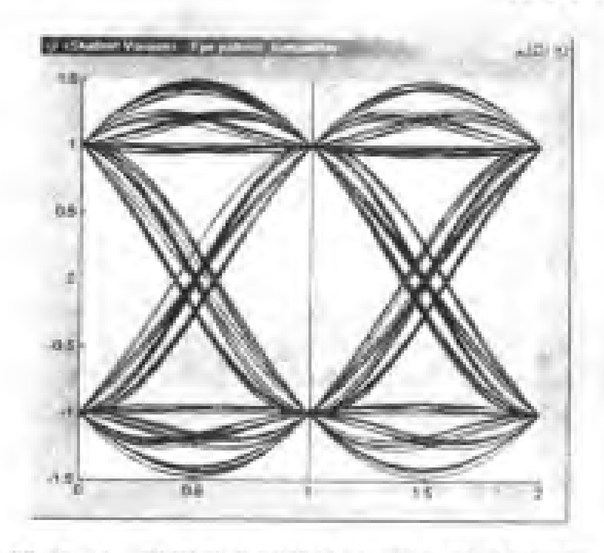


图 10.43 采用升余弦脉冲的 BPSK 发送端眼图

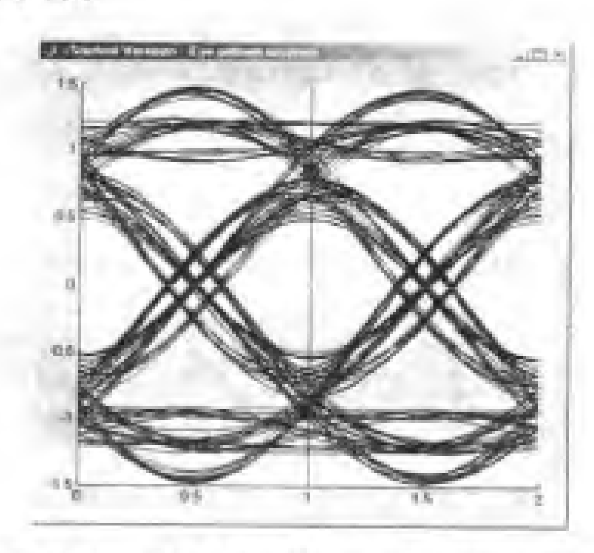


图 10.44 采用升余弦脉冲的 BPSK 接收端眼图

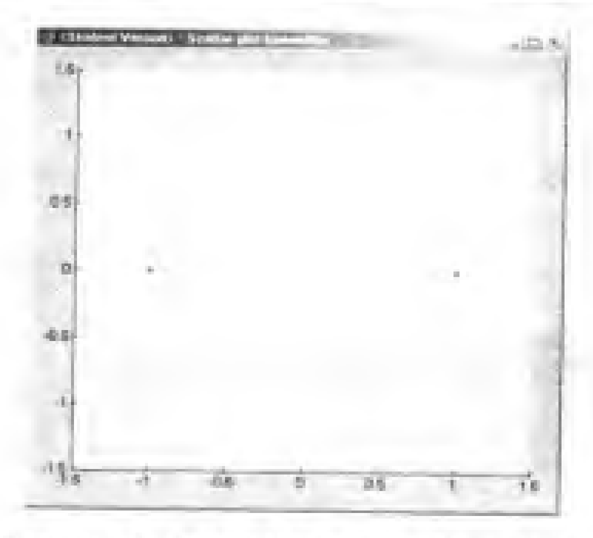


图 10.45 采用升余弦脉冲的 BPSK 发送端散布图

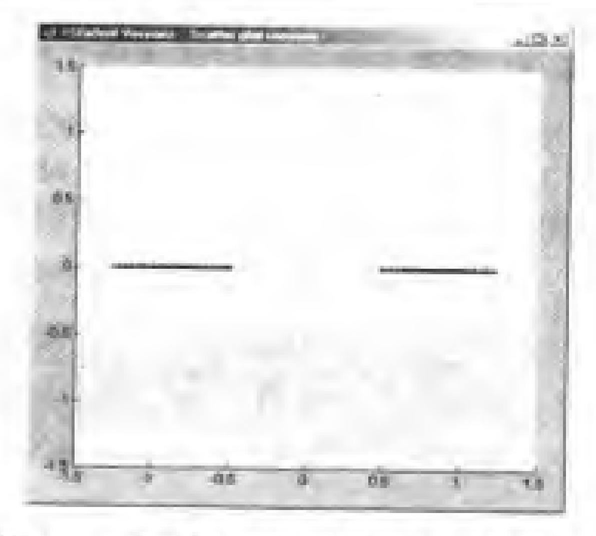


图 10.46 采用升余弦脉冲的 BPSK 接收端散布图



辅导题

辅导题 10.12 [采用升余弦脉冲的 BPSK]

1. 在发送端和接收端,零 ISI 条件和最大水平眼图开启条件满足吗?

2. 对检测来说,问题 1 的答案意味着什么? 考虑在接收端的眼图,若在发送端采用一个升余弦滤波器,而在接收端采用一个匹配滤波器,这样有意义吗?

3. 为什么信号空间矢端轨迹限制在 +1 和 -1 的范围内?

4. 与 NRZ 矩形脉冲相比,发射信号 $u(t)$ 如何改变? 在发送信号上带限的效果是什么?

5. 比较源信号 $a(t)$ 和宿信号 $v(t)$,它们之间存在误差吗 (考虑滤波器延时)?

题 解

1. 零 ISI 条件仅在发送端满足,这是由于当 $\alpha \neq 0$ 时两个升余弦脉冲 (匹配滤波器) 的级联会产生 ISI。另外,水平眼图开启不是最大。对于 $\alpha = 1$,会在发送端观察到最大水平眼图开启 (见辅导题 10.4)。

2. 在发送滤波器之后,检测能力最佳。但是在接收滤波器之后会见到 ISI,这就导致了垂直眼图开启度的下降,因此,升余弦脉冲与一个匹配接收滤波器不是最佳的。为了同时满足匹配滤波器条件,升余弦脉冲形状必须要在发送滤波器和接收滤波器之间均匀地等分割,这就使得在接收滤波器输出端的 SNR 最小并有零 ISI 条件,因此需要用到开根升余弦脉冲;这样,在接收端满足零 ISI 条件,而在发送端则不满足。

3. 这是由于升余弦脉冲形状所致,信号空间矢端轨迹的范围与眼图在垂直轴上的范围是一样的。

4. $u(t)$的包络不再是常数,带限会形成寄生幅度调制。矩形脉冲相移键控要求无限大的带宽,具有常数包络的已调信号总是要求无限大带宽,例如应用到频率调制信号。

5. 宿信号被发送和接收滤波器延时了,共延时 $2\times6\ T=12\ T$,差错数取决于噪声方差。

10.5.3 采用开根升余弦脉冲的二元相移键控

为了研究采用开根升余弦脉冲的 BPSK，可以选

BPSK > Root-RC

时域信号、眼图和散布图如图 10.47 至图 10.56 所示。

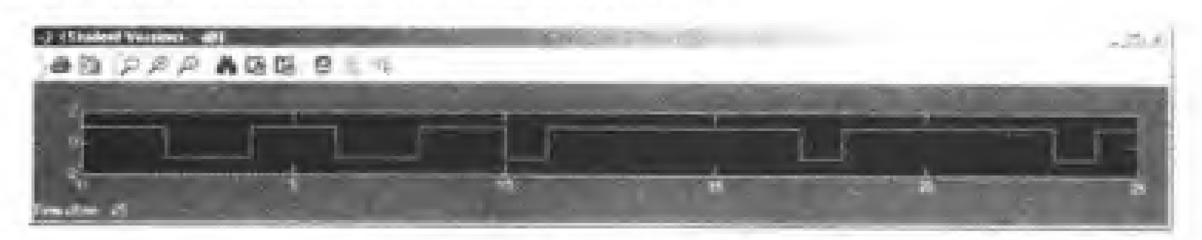


图 10.47 源信号 $a(t)$

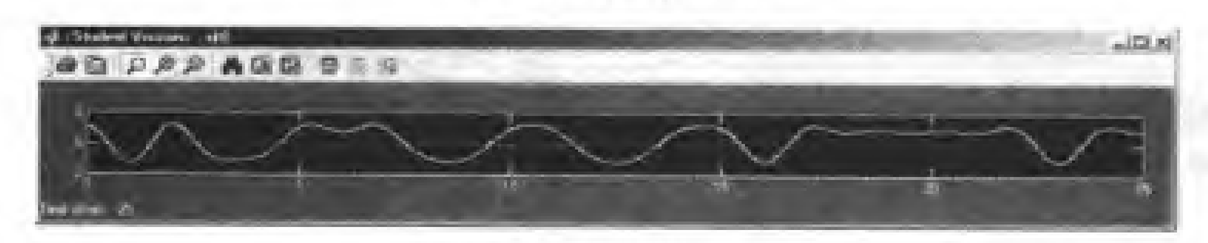


图 10.48 基带信号 $s(t)$

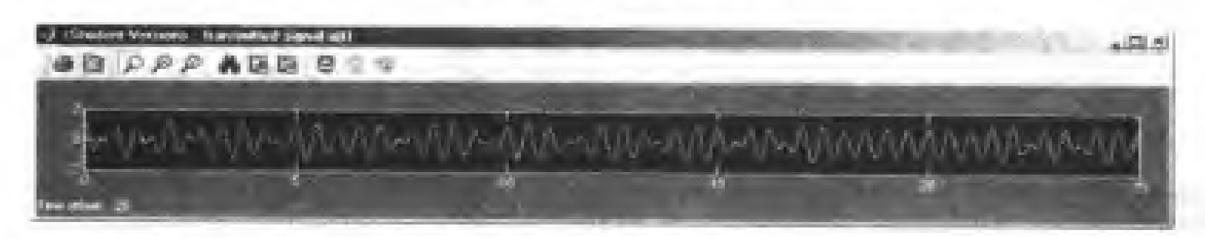


图 10.49 发送信号 $u(t)$

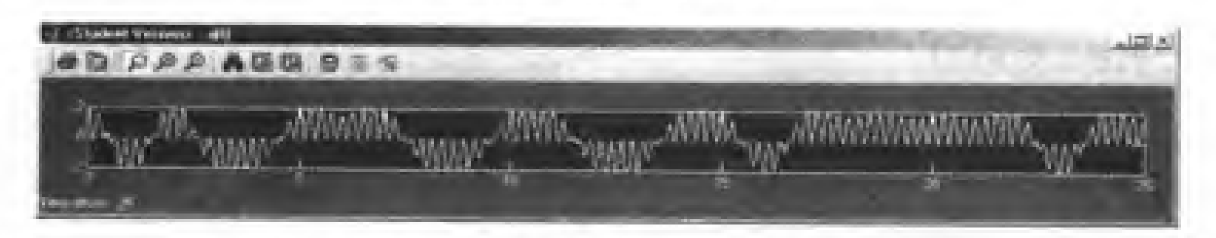


图 10.50 信号 $x(t)$（匹配滤波器输入端）

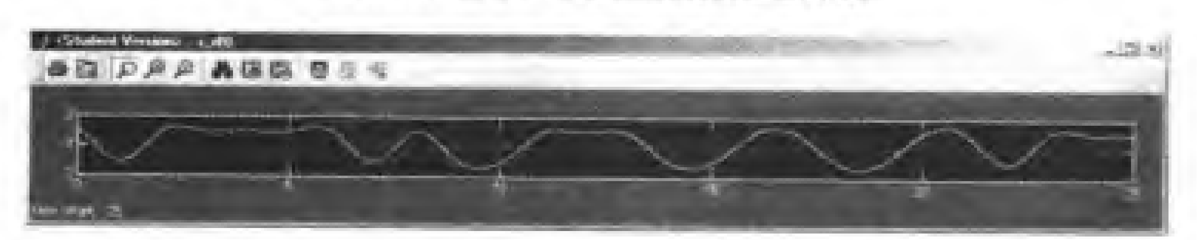


图 10.51 已解调基带信号 $s_r(t)$（匹配滤波器输出端）

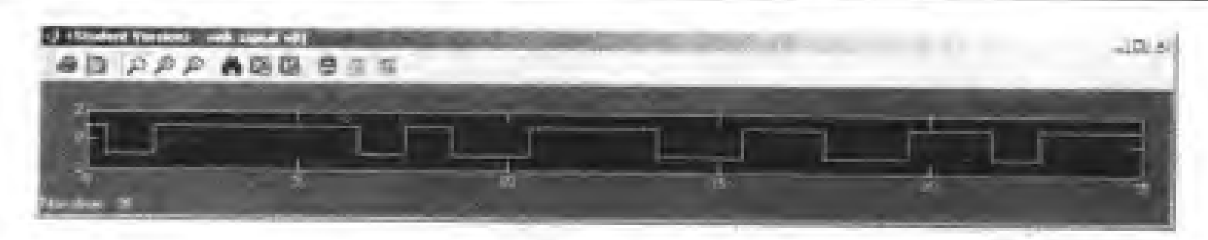


图 10.52　宿信号 $v(t)$

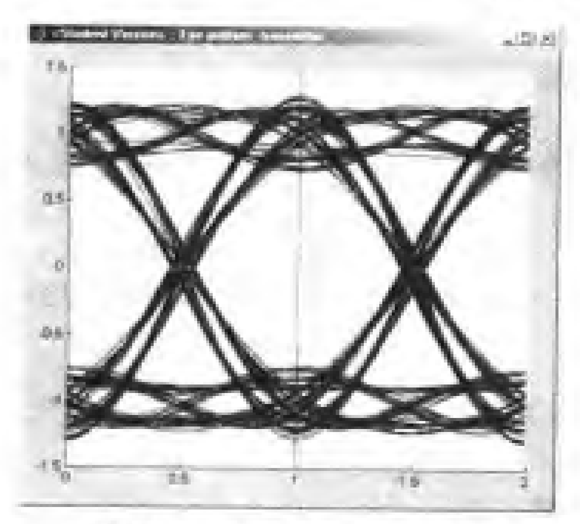


图 10.53　采用开根升余弦脉冲的 BPSK 发送端眼图

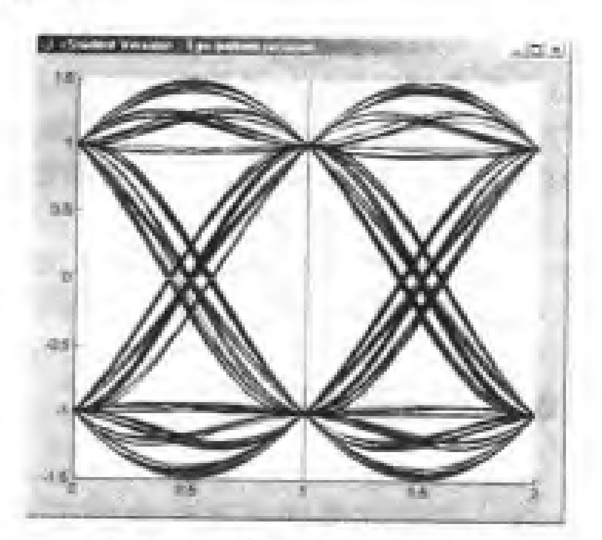


图 10.54　采用开根升余弦脉冲的 BPSK 接收端眼图

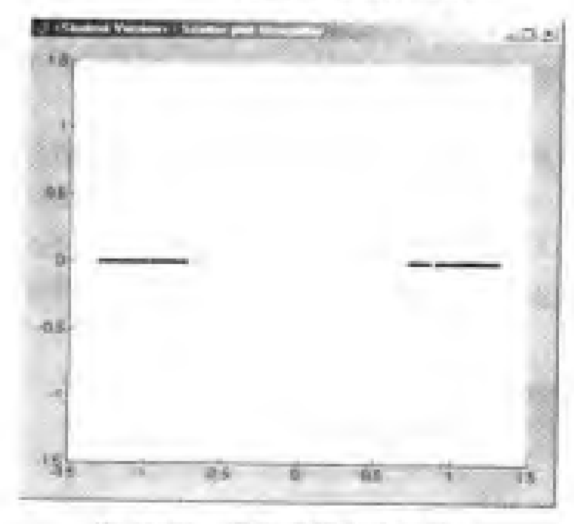


图 10.55　采用开根升余弦脉冲的BPSK发送端散布图



图 10.56　采用开根升余弦脉冲的BPSK接收端散布图

和前面对升余弦脉冲的情况相仿，可以完成相同的实验。值得注意的是，现在在匹配滤波器输出端满足零 ISI 条件，所以下面仅考虑开根升余弦发送端滤波器的有关情况。

仿真实验指南

对于 BPSK，我们已经考查过的几个方面对于其他调制方法来说也是可以考虑的。为此，

下面综合列出了最重要的几点，作为仿真的指南。下面各节将只提到新的或重要的一些方面：

1. 首先，选一个很小的噪声方差，稍后再考虑噪声影响。
2. 在发送端满足零 ISI 条件吗(使用眼图和信号空间图)?
3. 在发送端水平眼图开启是最大的吗(使用眼图)?
4. 在接收端满足零 ISI 条件吗?
5. 在接收端水平眼图开启是最大的吗?
6. 发送信号的包络是常数吗(说明脉冲形状和带限的影响)?
7. 在低通接收滤波器前后检查已解调信号。
8. 观察信号空间矢端轨迹并解释它的特性行为(零 ISI 条件)。可能有哪些转移? 信号空间覆盖的范围是什么?
9. 检查眼图(确定最佳采样时刻、眼图形状、垂直眼图开启度和加性噪声影响)。
10. 发生比特差错吗(采用不同的噪声方差检查对垂直眼图开启度和零 ISI 条件的影响)?

10.6　正交相移键控

10.6.1　采用开根升余弦脉冲的正交相移键控

检查用开根升余弦脉冲的 QPSK，可选：

QPSK > Root-RC

Simulink 模型如图 10.57 所示，分别双击蓝色方框 *Transmitter* 或 *Receiver*，会打开如图 10.58 和图 10.59所示的发送端和接收端模型。

打开发送端并启动仿真。值得注意的是，现在显示的是对两个正交分量的单独眼图，而这些与图 10.53 和图 10.54 中各自的 BPSK 眼图是一致的。两个正交分量的时域信号看起来像是各自的 BPSK 信号。对于无噪声传输时的 QPSK 信号，如图 10.60 至图 10.68 所示。在有噪信道中，接收端观察到的散布图看起来与图 10.67 类似。检查在正交分量中已调信号的包络和发送信号 $u(t)$的包络。

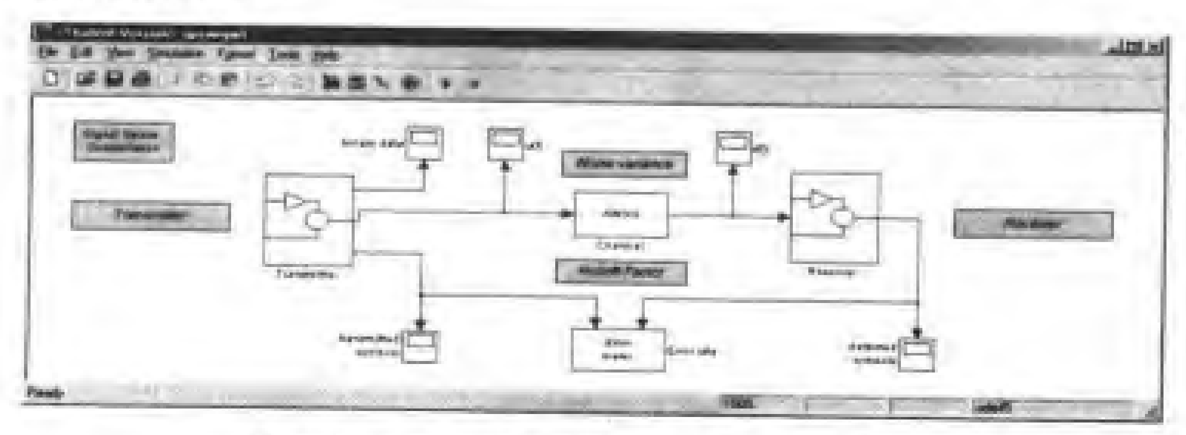


图 10.57　采用开根升余弦脉冲的 QPSK 的 Simulink 模型

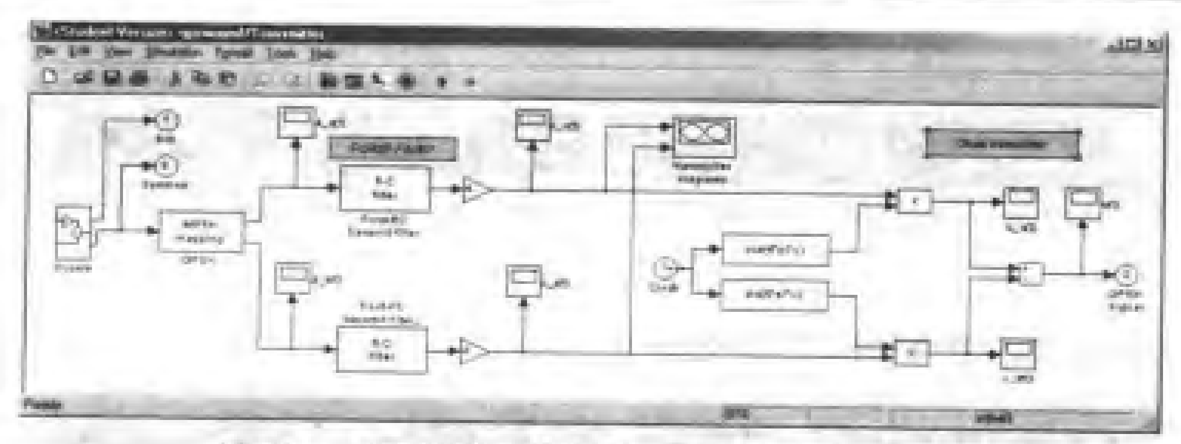


图 10.58 采用开根余弦脉冲的 QPSK 发送端的 Simulink 模型

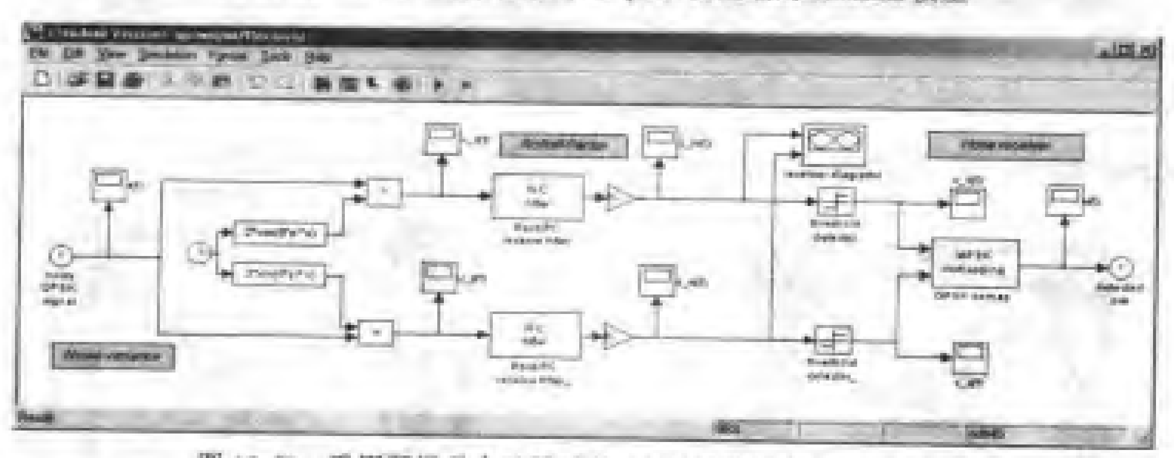


图 10.59 采用开根升余弦脉冲的 QPSK 接收端的 Simulink 模型

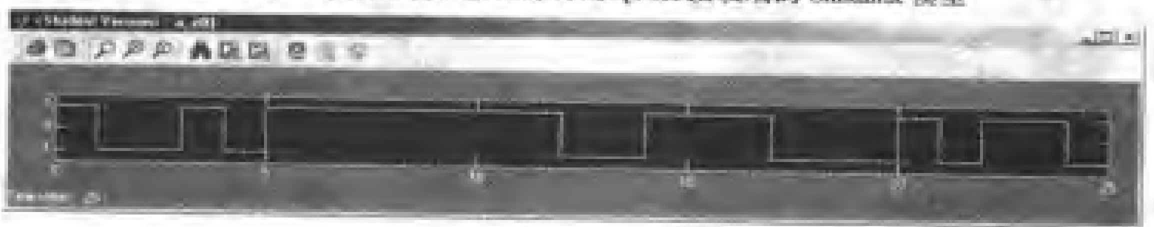


图 10.60 同相源信号 $a_c(t)$

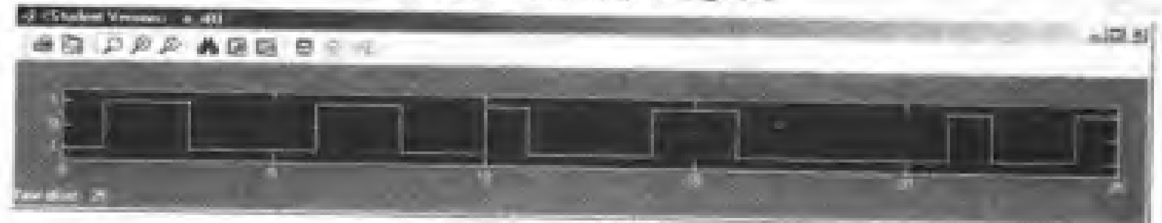


图 10.61 正交源信号 $a_s(t)$

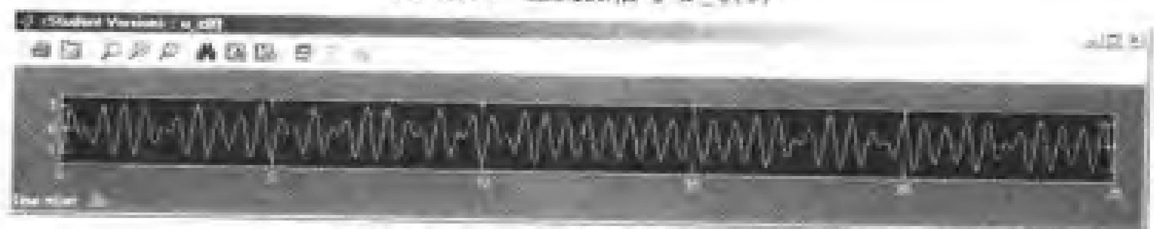


图 10.62 发送信号的同相分量 $u_c(t)$

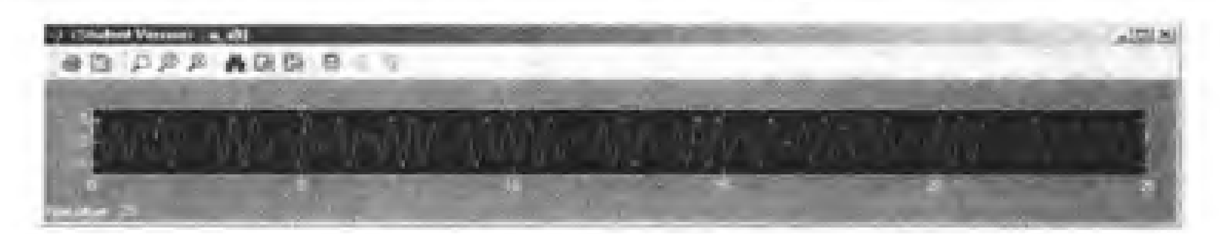


图 10.63 发送信号的正交分量 $u_s(t)$

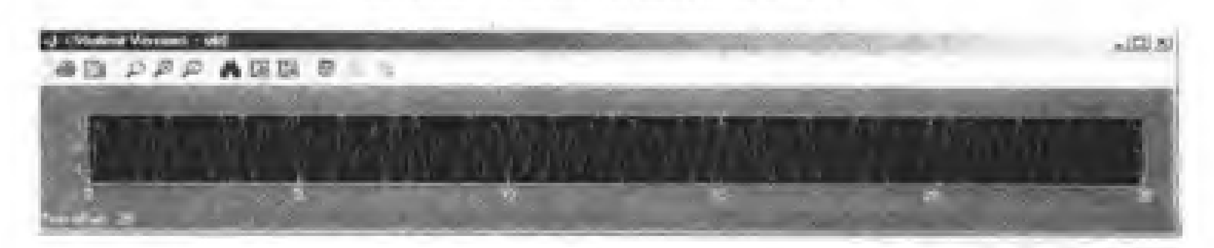


图 10.64 发送信号 $u(t)$

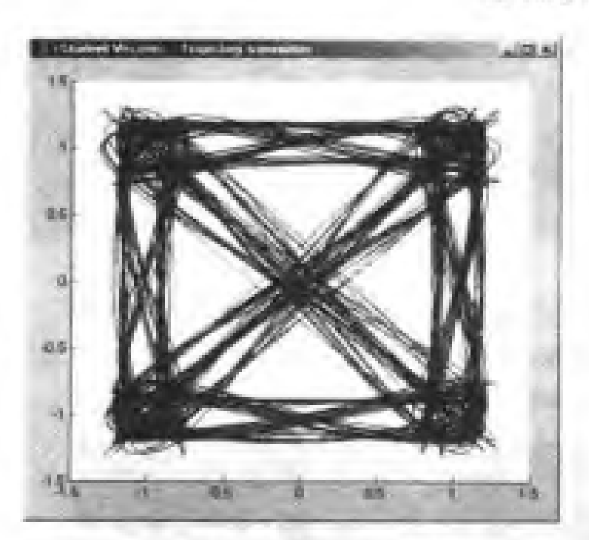


图 10.65 QPSK 发送端信号矢端轨迹图

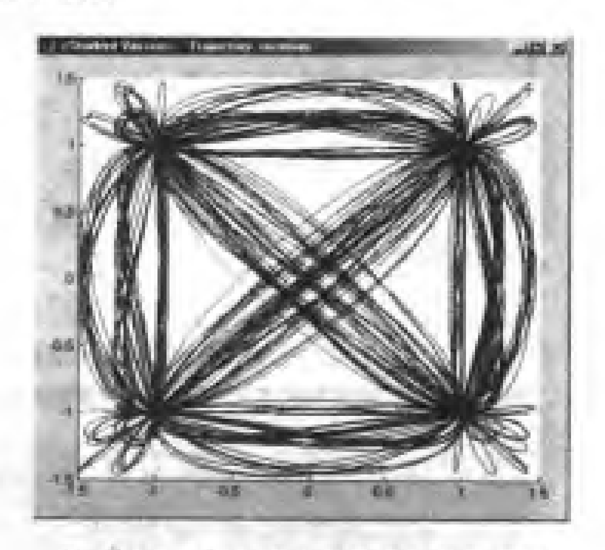


图 10.66 QPSK 接收端信号矢端轨迹图

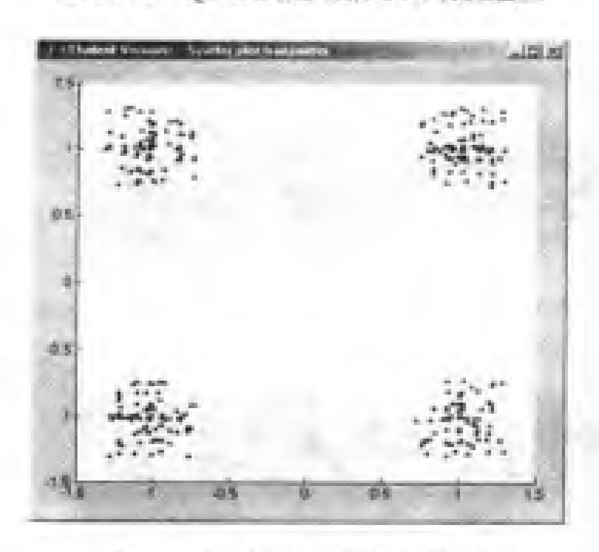


图 10.67 QPSK 发送端散布图

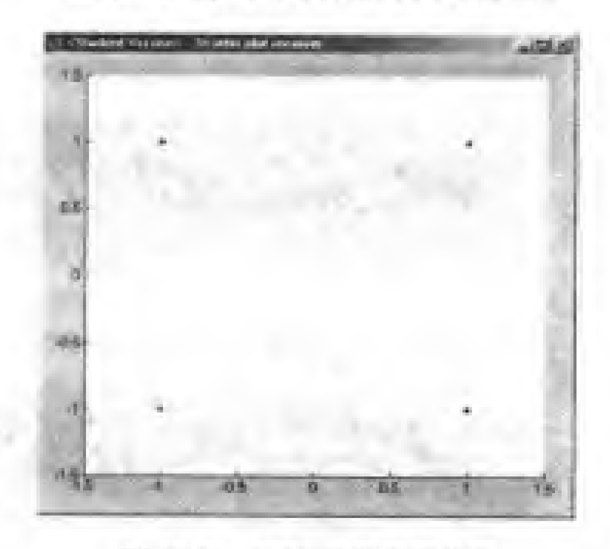


图 10.68 QPSK 接收端散布图

辅导题

辅导题 10.13　[采用开根升余弦脉冲的 QPSK：发送端]

1. 解释发送端的信号矢端轨迹图的特性行为。

2. 考虑到各自信号空间的转移，尽量发现在发送信号 $u(t)$ 包络中的衰落并说明理由（考虑发送滤波器延时为 $6T$）

题　解

1. 由于在发送端不满足零 ISI 条件，所以在 4 个 QPSK 信号空间星座点中矢端轨迹不呈现交叉节点。

2. 如果两个正交分量同时变化，则包络衰落。在这种情况下，信号矢端轨迹通过或接近原点完成对角线转移。由于带限，星座点之间的相移键控不是矩形，这就导致了包络衰落。降低包络衰落的一种简单方法是偏移 QPSK，将在 10.7 节讨论。

双击蓝色框 *Close transmitter*，然后再双击 *Receiver* 对接收端进行仿真。

辅导题

辅导题 10.14　[采用开根升余弦脉冲的 QPSK：接收端]

在眼图和信号空间图中观察到什么？

题　解

眼图：在采样时刻最大眼图开启，在采样时刻全部线都穿过两点。

信号矢端轨迹：若无噪声存在，轨迹线横穿 QPSK 星座点。

散布图：QPSK 星座点。

10.6.2　相位与频率偏移

现在要介绍用相位和频率失配对采用开根升余弦脉冲的 QPSK 进行解调。将噪声方差降低到最小值并打开 *Receiver* 模型。

相位偏移

双击接收端 Simulink 模型中的 cos 载波模块，显示出参数窗口。在两个正交分量中引入 $\varphi=\pi/8$ 的相位偏移，如图 10.69 和图 10.70 所示（u 记为频率），然后重新启动仿真。接收端的信号空间图和眼图见图 10.71 至图 10.74。

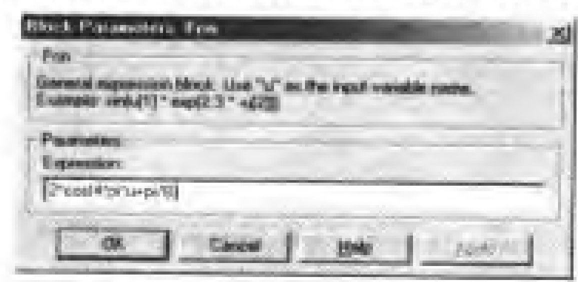


图 10.69　cos 载波参数窗

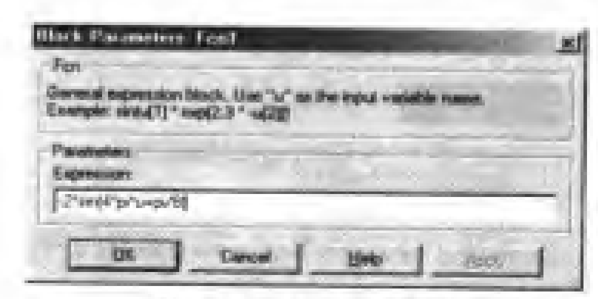


图 10.70　sin 载波参数窗

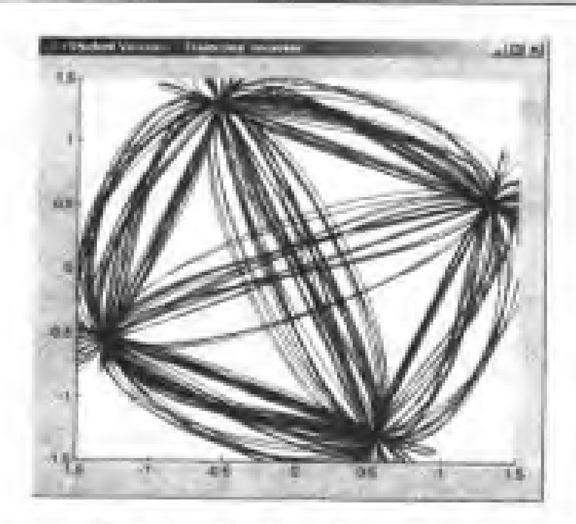


图 10.71　采用相位偏移 QPSK 的接收端信号矢端轨迹图

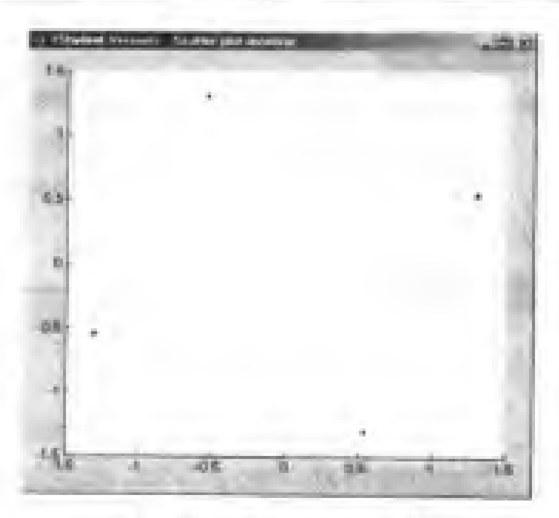


图 10.72　采用相位偏移 QPSK 的接收端散布图

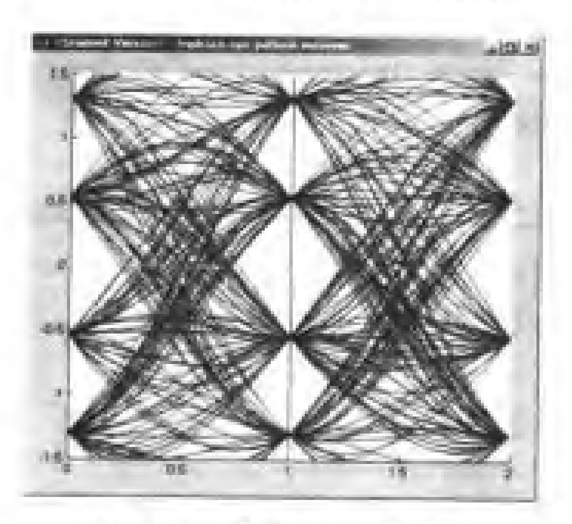


图 10.73　采用相位偏移 QPSK 在同相分量中的眼图

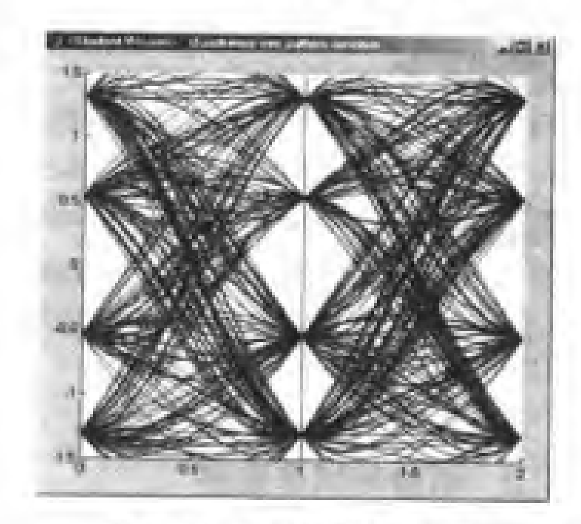


图 10.74　采用相位偏移 QPSK 在正交分量中的眼图

辅导题

辅导题 10.15　[采用相位偏移的 QPSK]

1. 在接收端散布图和信号矢端轨迹图上的影响是什么?
2. 如果没有噪声且 φ 对接收端未知，在 φ 的什么范围内可能实现无差错检测?
3. 解释在眼图中的效果。

题　解

1. 散布图和矢端轨迹图都旋转 φ。
2. $-\pi/4 < \varphi < \pi/4$。
3. 在两个正交分量中，现在在采样时刻可能有 4 个值，从图 10.72 的接收信号空间中可以看出这一点。

现在来看看如果在两个正交分量(cos 和 sin)中的解调载波不是准确相差 90°会产生什么后果。仅在 sin 载波分量中将相移重新置为 $\varphi = 0$(见图 10.75 和图 10.76)。

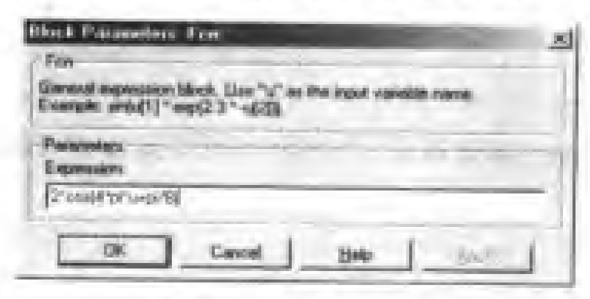


图 10.75 cos 载波参数窗



图 10.76 sin 载波参数窗

产生的有关图见图 10.77 至图 10.80。

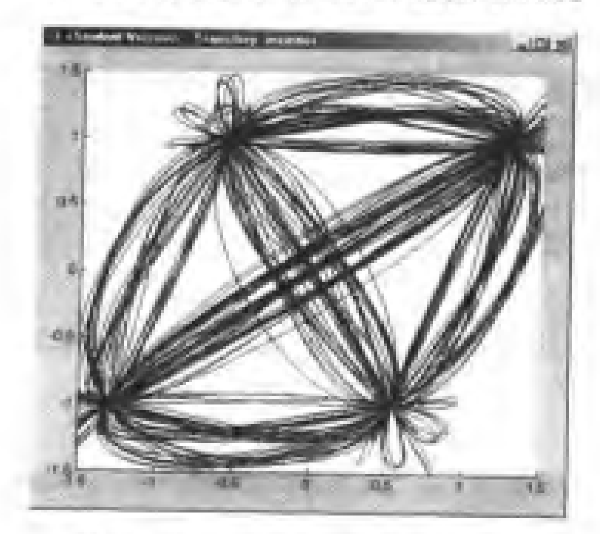


图 10.77 cos 分量有相位偏移的 QPSK 接收端信号矢端轨迹图

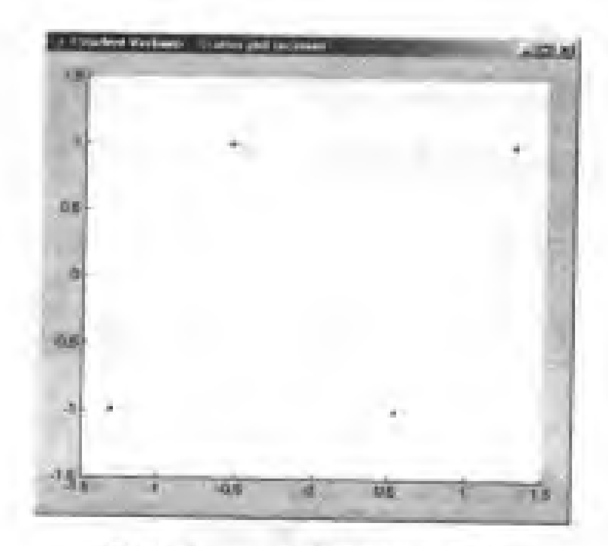


图 10.78 cos 分量有相位偏移的 QPSK 接收端散布图

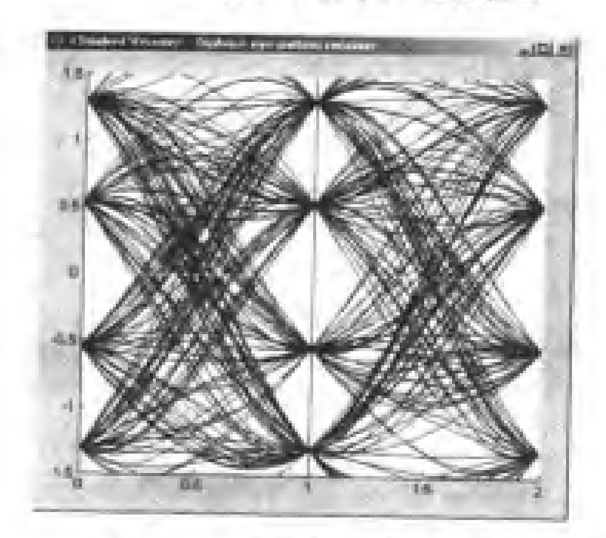


图 10.79 cos 分量有相位偏移的 QPSK 在同相分量中的眼图

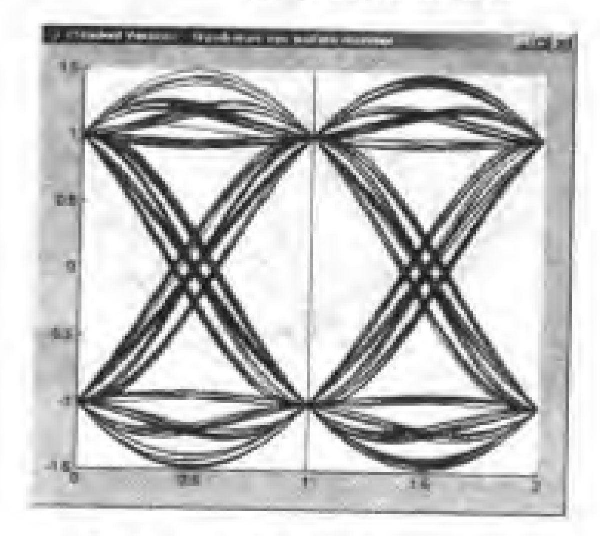


图 10.80 cos 分量有相位偏移的 QPSK 在正交分量中的眼图

辅导题

辅导题 10.16　[相位失配 QPSK]

描述在散布图中的效果。

题　解

在 sin 分量中，仅在采样时刻有正确的 ±1 值。在 cos 分量中，仍然可以观察到 4 个可能的值。因此，接收信号空间星座图是失真的。

频率偏移

为了研究频率偏移，需要按图 10.81 和图 10.82 修改解调载波，图 10.83 至图 10.86 给出了所得到的信号空间图和眼图。



图 10.81　cos 载波参数窗

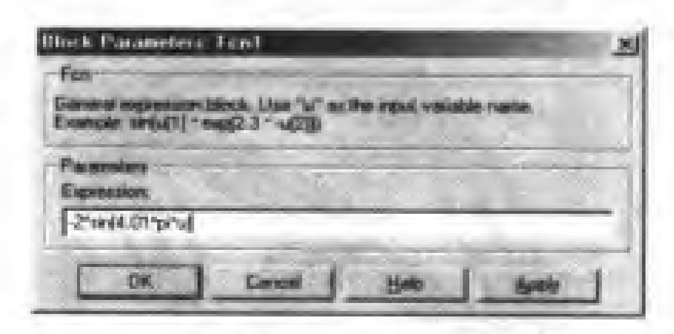


图 10.82　sin 载波参数窗

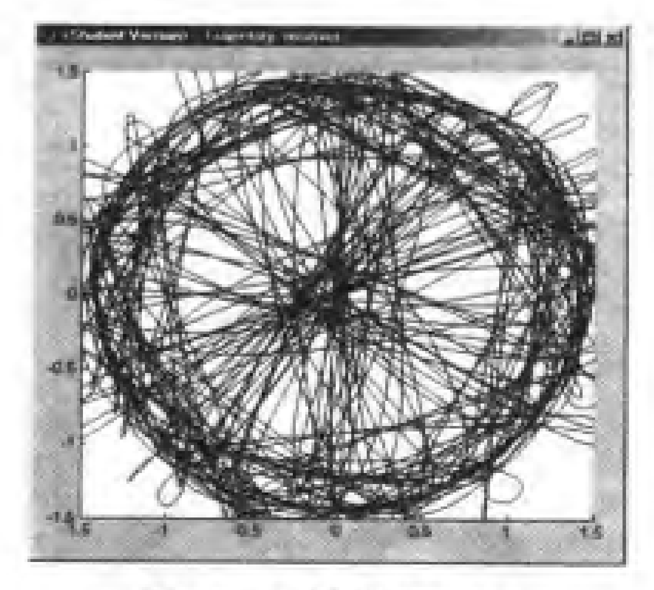


图 10.83　频率偏移 QPSK 接收端信号矢端轨迹图

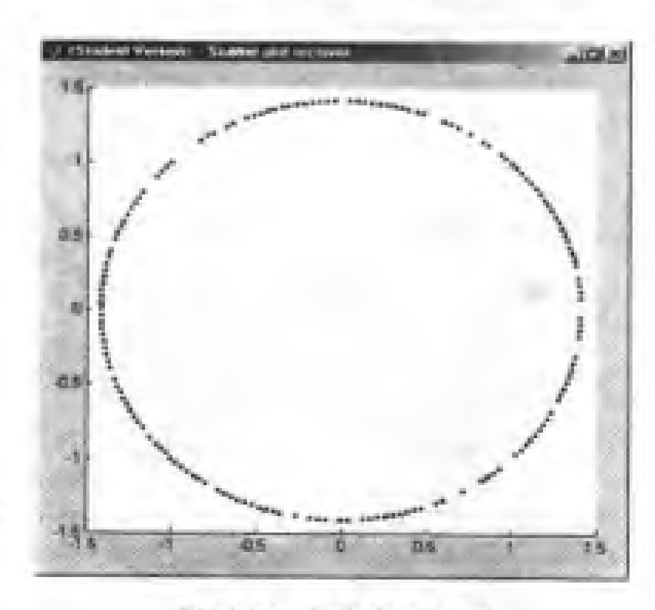


图 10.84　频率偏移 QPSK 接收端散布图

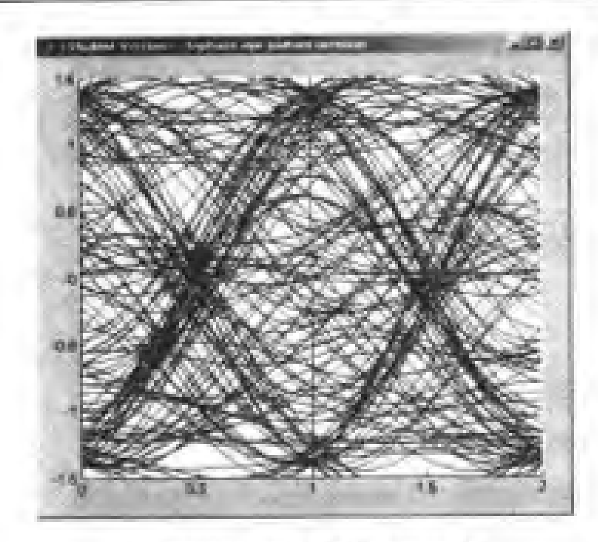


图 10.85　频率偏移 QPSK 在同相分量中的眼图

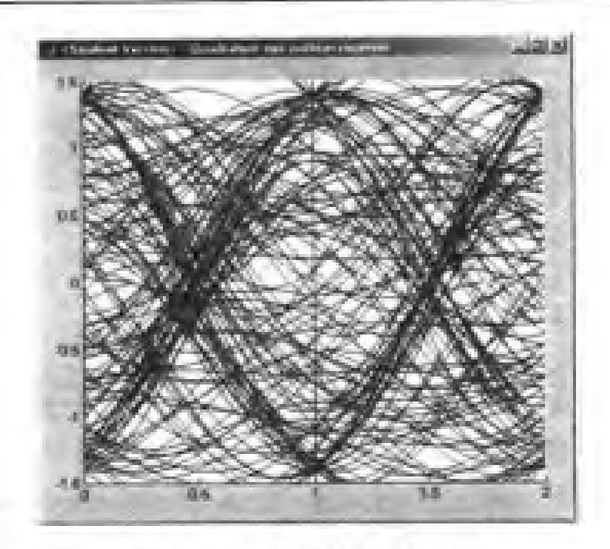


图 10.86　频率偏移 QPSK 在正交分量中的眼图

辅导题

辅导题 10.17　[频率偏移 QPSK]

描述眼图和信号空间图。无差错检测可能吗?

题　解

接收信号空间星座按照频率偏移 Δf 旋转，不可能进行检测。由矢端轨迹图和眼图的混沌行为也能得到这个结论。

10.7　偏移QPSK

10.7.1　理论

在辅导题 10.13 中曾遇到过QPSK已调信号的包络衰落问题。如果两个接续的符号关于信号空间星座图是对称的，那么信号矢端轨迹完成的是通过原点的对角线转移。这就形成了发送信号的幅度衰落。这样，在转移过程中发送信号对由放大器施加的非线性就很灵敏，因此很多应用都要求发送信号有接近常数的包络。避免深度衰落的一种简单方法是应用偏移QPSK。

在偏移QPSK中，QPSK信号的正交分量相对于同相分量延时 $T/2$。所以，同相分量和正交分量不再能同时改变，从而避免了矢端轨迹的对角线转移。

10.7.2　实验

现考察采用开根升余弦脉冲的偏移 QPSK，选取：

Offset-QPSK > Root-RC

其基本 Simulink 模型与图 10.57 所示的 QPSK 模型是相同的，惟一的变化是在发送端的正交分量延时 $T/2$(见图 10.87)；在接收端将同相分量延时 $T/2$，以恢复到 QPSK 的星座(见图 10.88)。图 10.89 至图 10.99 分别给出了偏移 QPSK 的有关信号和图形。

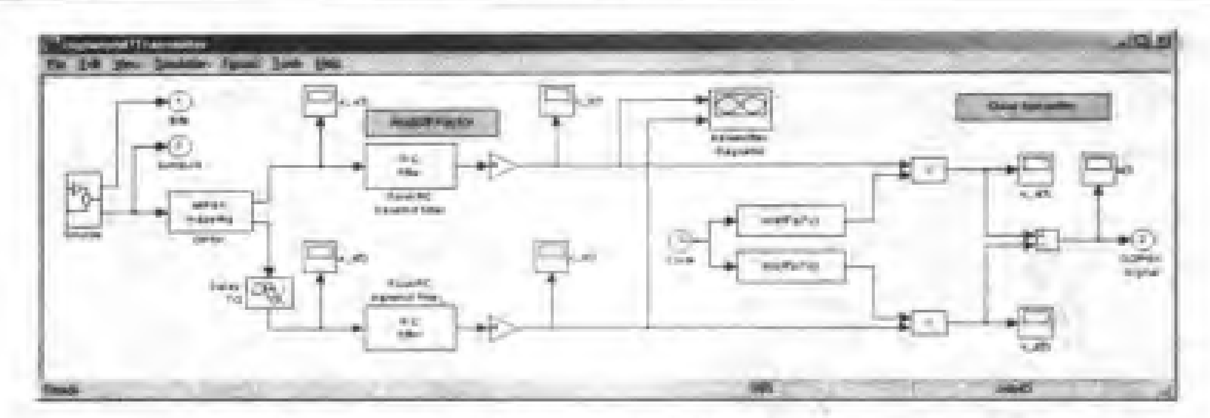


图 10.87 采用开根升余弦的偏移 QPSK 的发送端 Simulink 模型

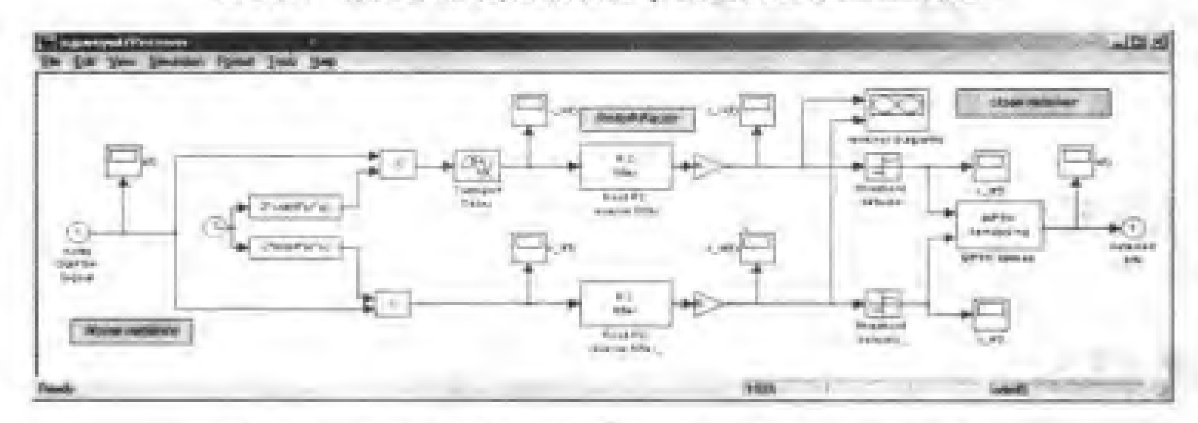


图 10.88 采用开根升余弦的偏移 QPSK 的接收端 Simulink 模型

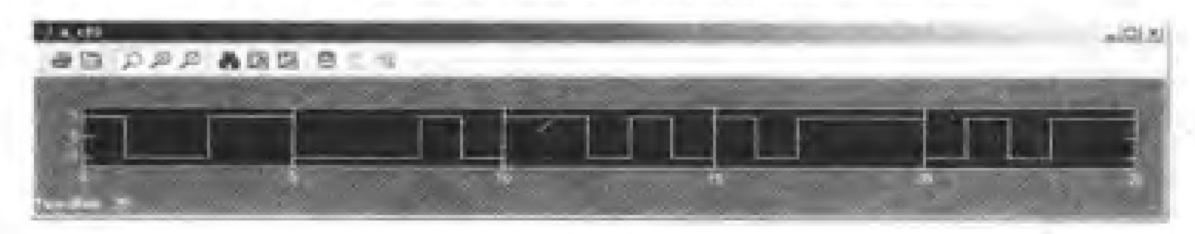


图 10.89 同相源信号 $a_c(t)$



图 10.90 正交源信号 $a_s(t)$

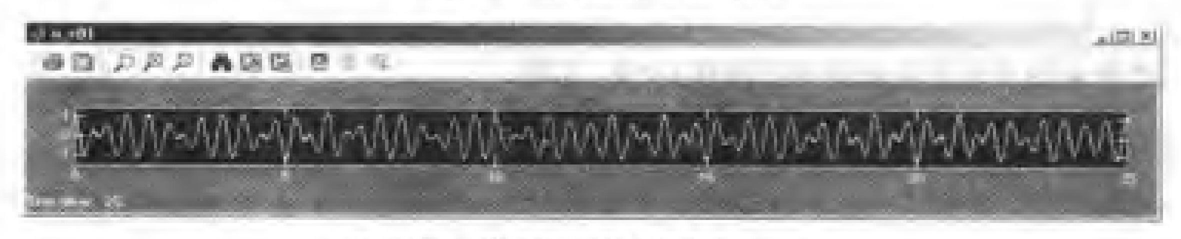


图 10.91 发送信号的同相分量 $u_c(t)$

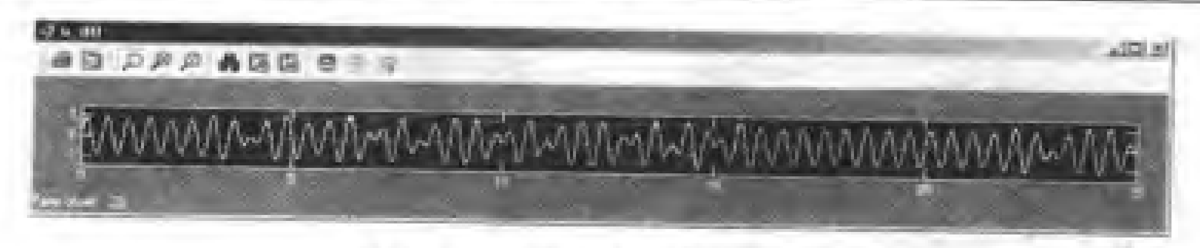


图 10.92 发送信号的正交分量 u_s(t)

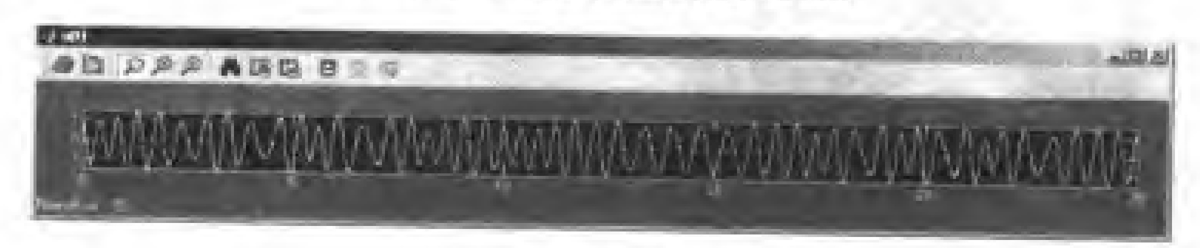


图 10.93 发送信号 u(t)

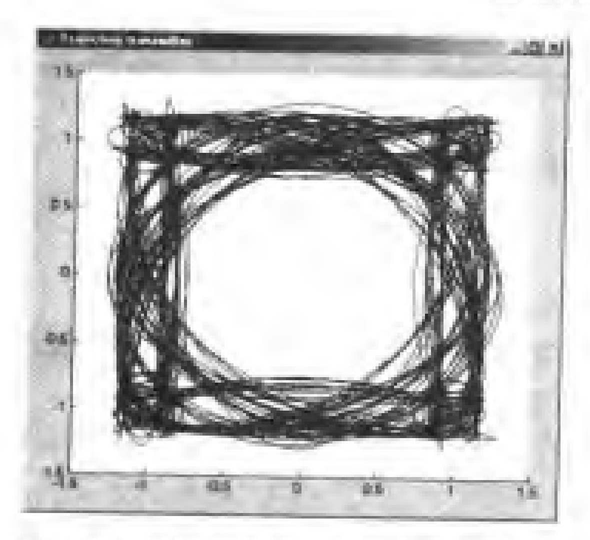


图 10.94 偏移 QPSK 发送端信号矢端轨迹图

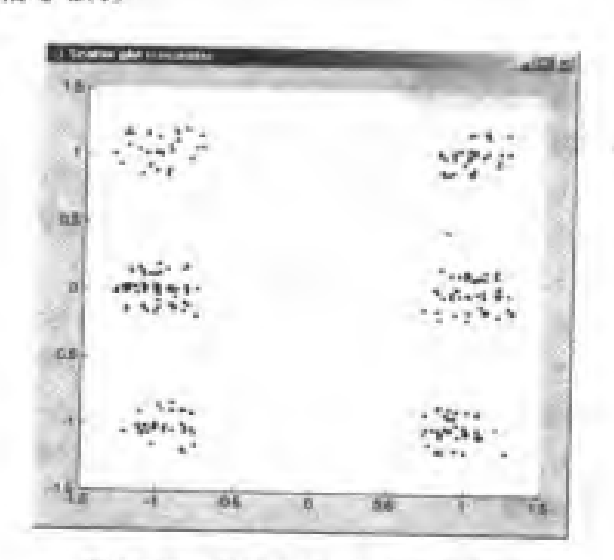


图 10.95 偏移 QPSK 发送端散布图

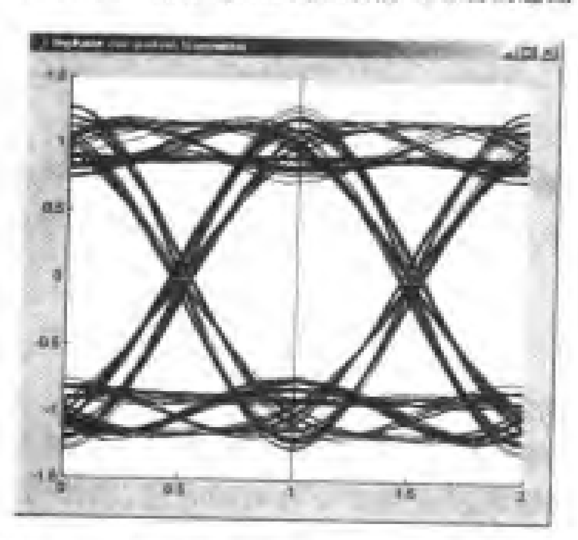


图 10.96 偏移 QPSK 发送端在同相分量中的眼图

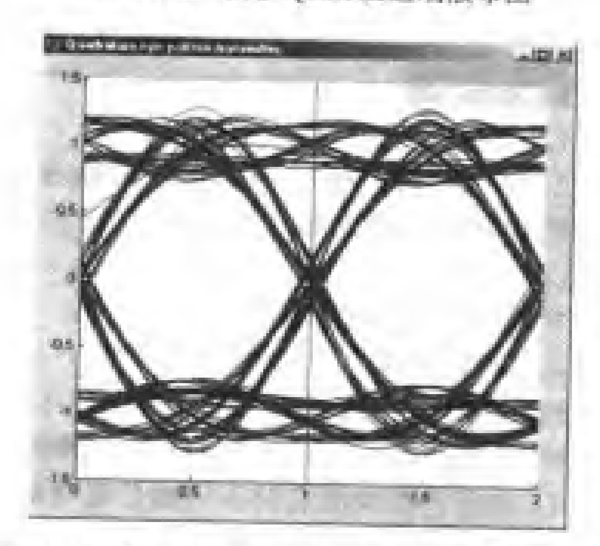


图 10.97 偏移 QPSK 发送端在正交分量中的眼图

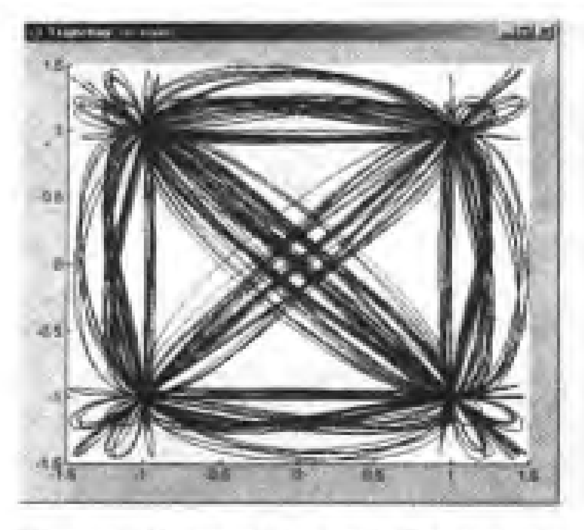


图 10.98 偏移 QPSK 接收端信号矢端轨迹图



图 10.99 偏移 QPSK 接收端散布图

辅导题

辅导题 10.18 [偏移 QPSK]

1. QPSK 和偏移 QPSK 的信号空间星座图有差异吗?

2. 解释在脉冲成形之后发送端散布图的结果。

3. 两个正交分量的眼图差别如何?

4. 偏移 QPSK 和 QPSK 的发送端信号矢端轨迹有什么不同?

5. 信号 $a_c(t)$、$a_s(t)$、$u_c(t)$和 $u_s(t)$与图 10.60 至图 10.63 中各自的 QPSK 信号差别怎样? 注意 $a_c(t)$和 $a_s(t)$永远不可能同时变化。

6. 检查发送信号 $u(t)$,并与对应的图 10.64 中的 QPSK 信号比较包络衰落。说明它们之间的差异。

7. 为什么在接收端还观察到信号矢端轨迹的对角线转移? 利用 Simulink 接收端模型来阐明。

题 解

1. 没有。

2. 由于延时,相对于同相分量而言在正交分量中最佳采样时刻时移了 $T/2$。Simulink 散布图在同相分量的最佳采样时刻采样。因此,在正交分量中观察到多于两个信号空间集束。

3. 正交分量眼图时移了 $T/2$。

4. 偏移 QPSK 不存在对角线转移。

5. 在偏移 QPSK 中,$a_s(t)$和 $u_s(t)$时移了 $T/2$。

6. 因为在两个正交分量中数据不能同时变化,所以偏移 QPSK 有比较小的包络衰落。

7. 在接收滤波器之前将同相分量时移 $T/2$,以补偿在发送端正交分量的时移,因此可以看到与 QPSK 相同的信号矢端轨迹。

10.8　最小频移键控

10.8.1　理论

在 7.5 节中介绍过二元频移键控(FSK)，并导出了两个信号波形之间的互相关系数 γ_{mn} 是频率间隔 Δf 的函数(见图 7.25)。已经证明，频率间隔 $1/2T$ 是正交信号波形的最小频率分隔，因此具有 $\Delta f = 1/2T$ 的二元 FSK 称为最小频移键控(MSK)。

一般来说，FSK 是一种非线性数字调制方法。然而，由于正交性，MSK 也可以看成是一种线性调制方法：MSK 等效于具有余弦脉冲和差分预编码的偏移 QPSK。偏移 QPSK 的载波频率是在式(7.5.2)的发送信号中两个频率 $f_1 = f_c$ 和 $f_2 = f_c + \Delta f$ 之间的中心频率：

$$f_{c,o} = f_c + \frac{\Delta f}{2} \tag{10.8.1}$$

一个 MSK 调制器的偏移 QPSK 表示如图 10.100 所示。为了确保发送信号有连续相位，差分预编码是必要的。有关这方面的更为详尽的说明可参阅参考文献[3]。每隔一个比特分别调制 cos 或 sin 分量，所以两个正交分量之间的延时是 T。由于 MSK 是一种载波频率调制，发送信号 $u(t)$的包络是一个常数。利用余弦脉冲可以得到：

$$g_T(t) = \begin{cases} \sqrt{\dfrac{4\mathcal{E}}{T}}\cos\left(\dfrac{\pi t}{2T}\right), & -T \leqslant t \leqslant T \\ 0, & \text{其余 } t \end{cases} \tag{10.8.2}$$

10.8.2　实验

从主菜单上选取 MSK 并启动仿真。基本的 Simulink 模型与图 10.57 所示的 QPSK 模型是类似的。MSK 发送端模型如图 10.101 所示。图 10.102 至图 10.112 给出了 MSK 的有关图形。

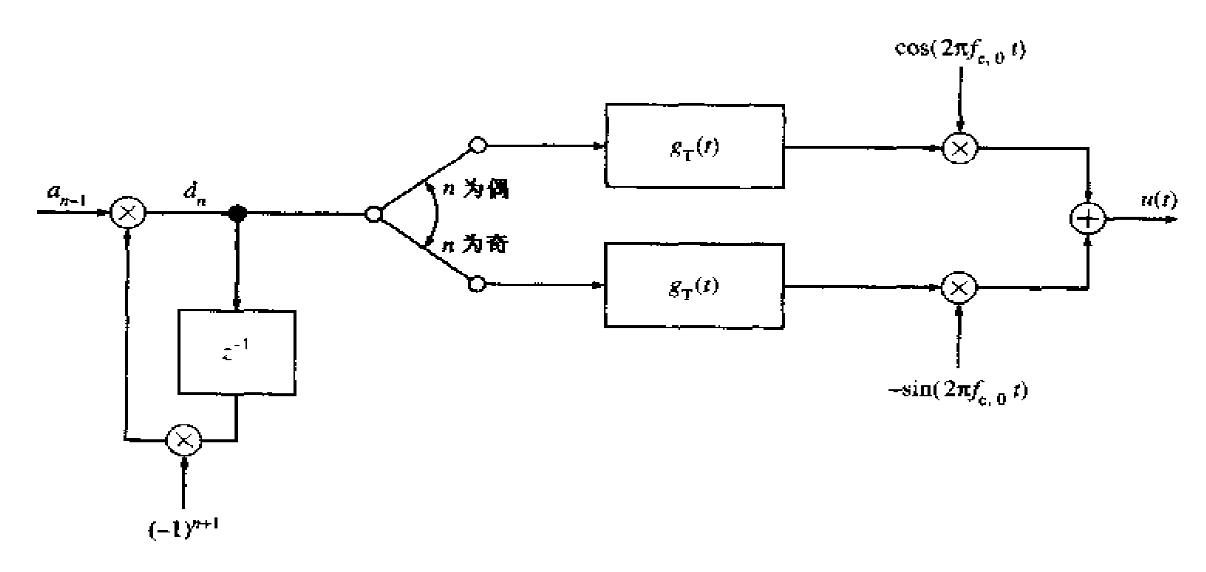


图 10.100　MSK 调制器的偏移 QPSK 表示

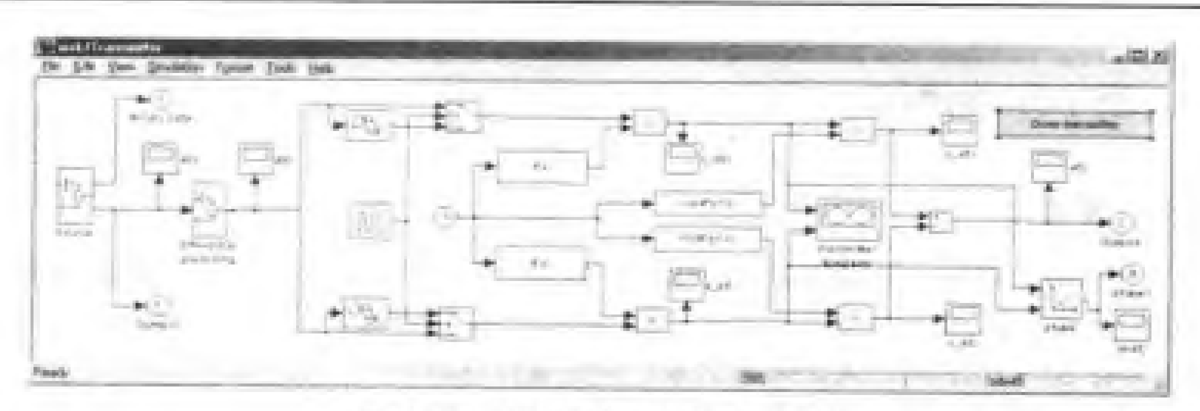


图 10.101　MSK 发送端的 Simulink 模型

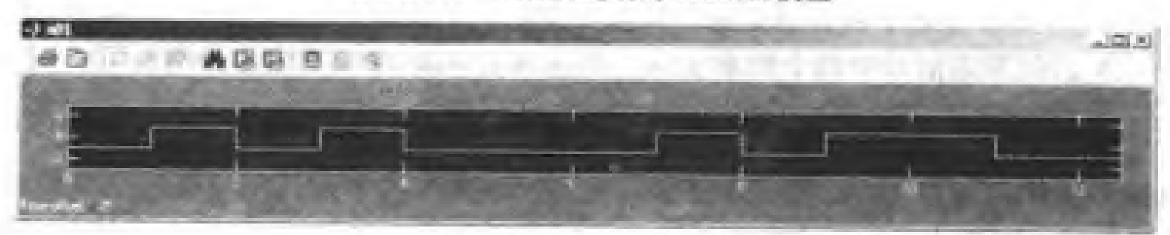


图 10.102　源信号 $a(t)$

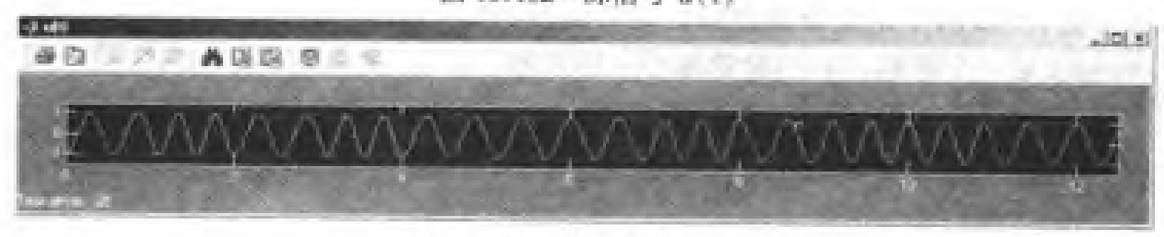


图 10.103　发送信号 $u(t)$

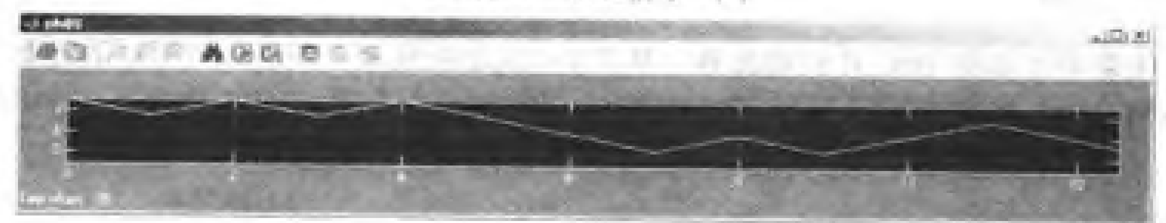


图 10.104　信号矢端轨迹相位

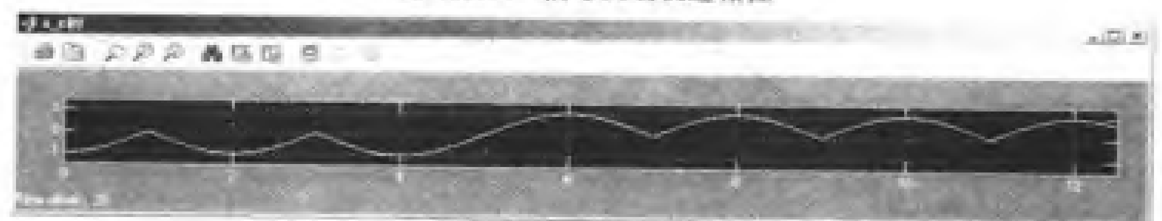


图 10.105　同相基带信号 $s_c(t)$

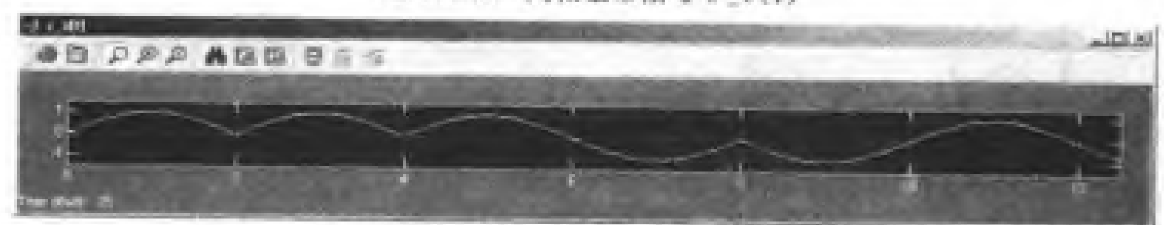


图 10.106　正交基带信号 $s_s(t)$

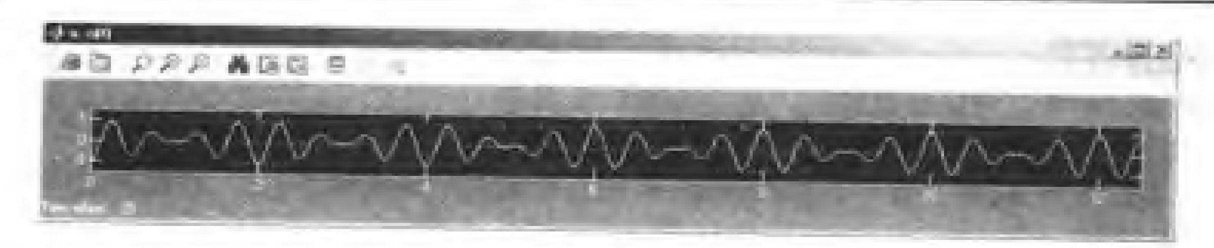


图 10.107　发送信号的同相分量 $u_c(t)$

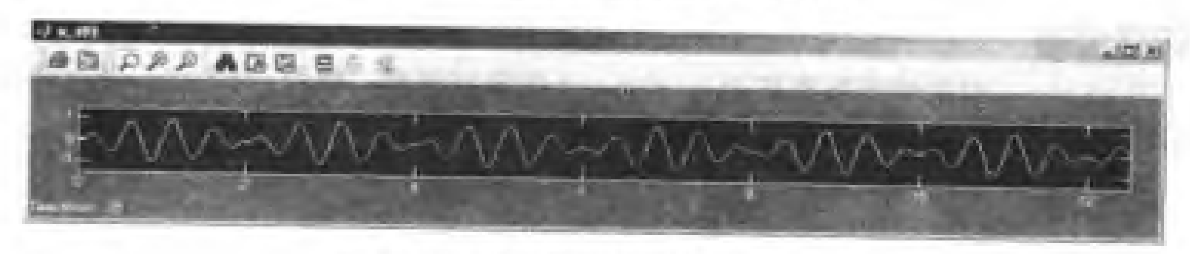


图 10.108　发送信号的正交分量 $u_s(t)$

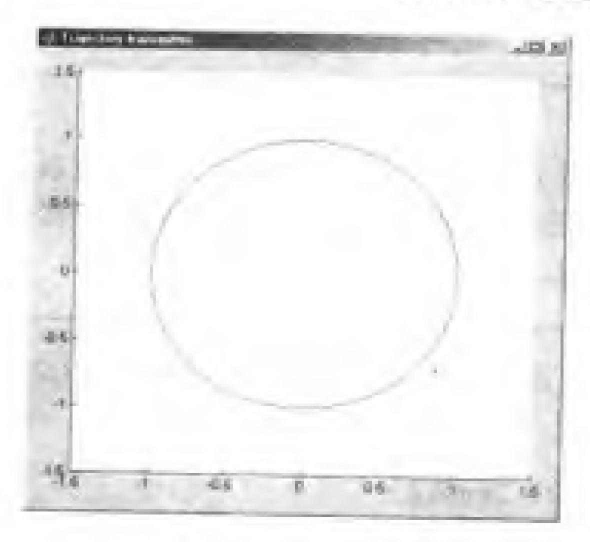


图 10.109　MSK 发送端信号矢端轨迹图

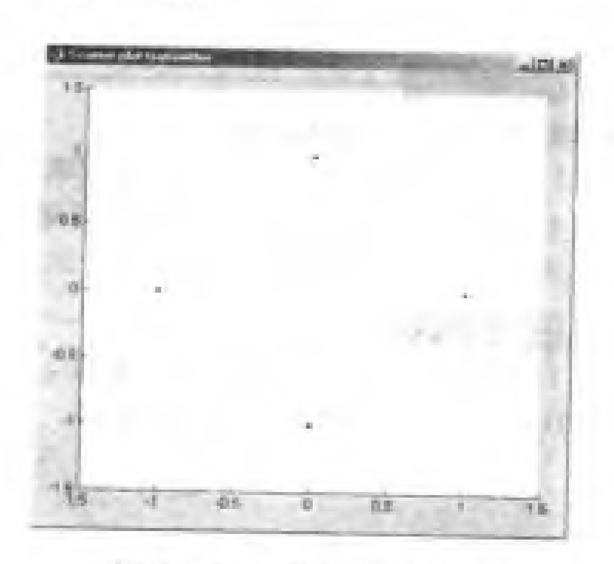


图 10.110　MSK 发送端散布图

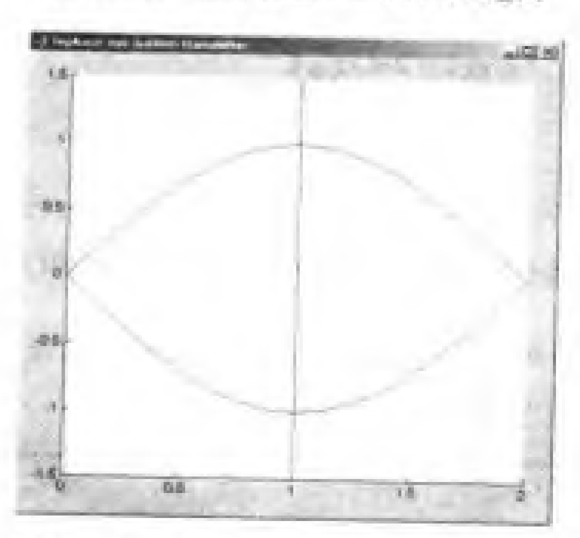


图 10.111　MSK 发送端在同相分量中的眼图

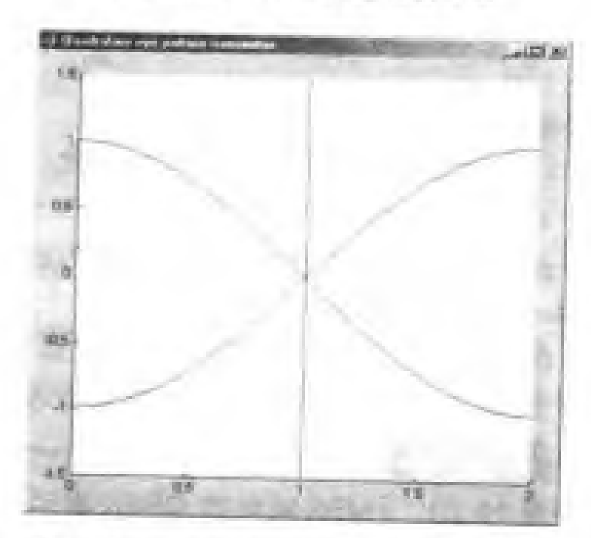


图 10.112　MSK 发送端在正交分量中的眼图

辅导题

辅导题 10.19 [二元频移键控]

把源信号 $a(t)$的窗放在发送信号 $u(t)$窗的上面。

1. 我们观察到,MSK 是一种频率调制,即信息是包含在载波频率中的,并且发送信号$u(t)$的包络是常数。

2. 在信号矢端轨迹中常数包络是如何表现的?

题解

1. 当 $a(t)=1$ 时的载波频率高于 $a(t)=-1$ 的载波频率。然而,由于选的是正交信号的最小频率间隔,因此不易看出频率上的差别。

2. 矢端轨迹是一个圆。

在仿真的一次重新启动之后,将矢端轨迹窗拖到屏幕当中以免其他窗挡住它。重新启动仿真并观察信号矢端轨迹。

辅导题

辅导题 10.20 [最小频移键控]

1. 结合源信号 $a(t)$解释信号矢端轨迹的特性行为(轨迹是在第 2 个源比特开始显示的)。

2. 观察轨迹相位 *phi*(t),并将它与源信号 $a(t)$进行比较。在一个周期 T 内的相移为多少? 要注意是连续相位,它是由于差分预编码得到的。

题解

1. 轨迹在一个圆上以每个 T 相移 $\pm\pi/2$ 旋转。这是根据 MSK 的频率间隔 $\Delta f=1/2T$ 和式(10.8.1),按照式(7.5.2)的发送信号

$$
\begin{aligned}
u_m(t) &= \sqrt{\frac{2\mathcal{E}}{T}}\cos(2\pi f_c t + 2\pi m\Delta f t) \\
&= \sqrt{\frac{2\mathcal{E}}{T}}\cos\left(2\pi f_{c,o} t + 2\pi(1-2m)\frac{\Delta f}{2}t\right) \\
&= \sqrt{\frac{2\mathcal{E}}{T}}\cos\left(2\pi f_{c,o} t + \pi(1-2m)\frac{t}{2T}\right)
\end{aligned}
$$

而得出的。

2. 相位以每 T 相移 $\pm\pi/2$ 线性变化。

现在考察如何将 MSK 表示成偏移 QPSK。双击蓝色方框 *Transmitter* 打开如图 10.101 所示的 Simulink 发送端模型。

在脉冲成形之后,观察两个正交分量中的基带信号 $s_c(t)$和 $s_s(t)$。cos 脉冲确保了发送信号 $u(t)$具有恒定包络,或者等效为矢端轨迹是一个圆。

辅导题

辅导题 10.21 [MSK 的偏移 QPSK 表示]

1. 在正交分量中,cos 脉冲的时间间距是多少? 正交分量相对于同相分量的延时是多少?

2. 解释为什么在发送端散布图中出现 4 个点。

题　解

1. 间距：$2T$，正交分量延时：T。

2. 以采样率 $1/T$ 对信号矢端轨迹采样，由图 10.105 和图 10.106 可见 $s_c(2kT)=\pm 1$ 和 $s_s(2kT)=0$，而 $s_c((2k+1)T)=0$ 和 $s_s((2k+1)T)=\pm 1$。因此，在 $2kT$ 对轨迹采样产生散布图中的一个同相点，而在 $(2k+1)T$ 对轨迹采样产生一个正交点。

MSK 能够以和偏移 QPSK 相同的方式进行相干解调，再紧跟一个差分解码器，因此在此不必详尽地处理解调问题。

10.9　16 电平正交幅度键控

10.9.1　采用开根升余弦脉冲的 16 电平正交幅度键控(16-QAM)

考察采用开根升余弦脉冲的 16-QAM：

16-QAM > Root-RC

Simulink 模型和发送端模型分别如图 10.113 和图 10.114 所示。在 16-QAM 中，将 $b=4$ 比特映射在一个信号空间星座点。通过双击蓝色方框 *Signal Space Constellation* 能够看到这个信号空间星座图如图 10.115 所示。

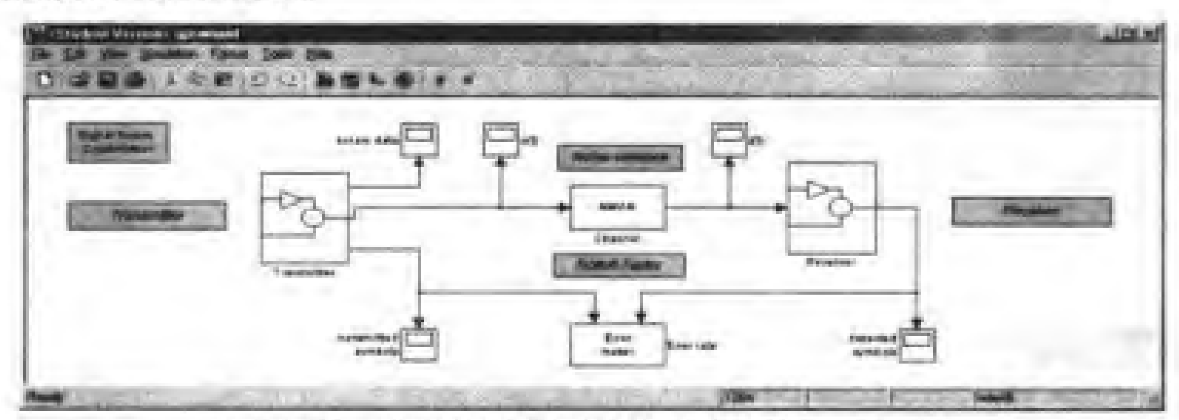


图 10.113　采用开根升余弦脉冲的 16-QAM 的 Simulink 模型

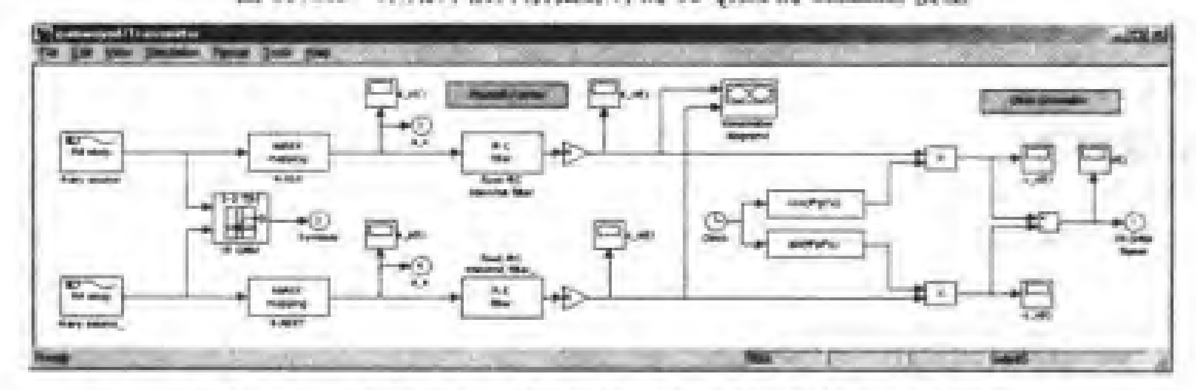


图 10.114　采用开根升余弦脉冲的 16-QAM 发送端的 Simulink 模型

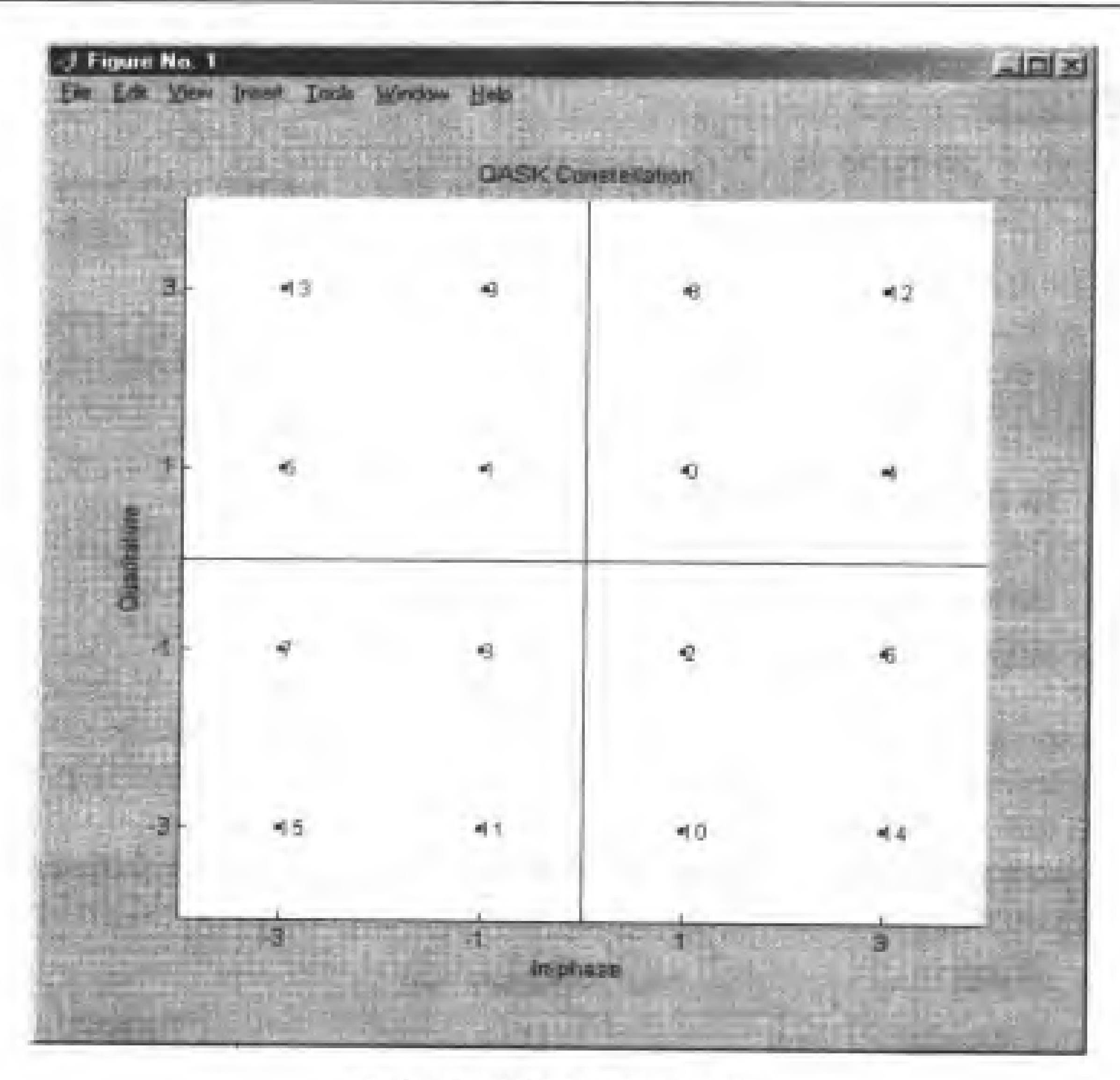


图 10.115　16-QAM 信号空间星座图

辅导题

辅导题 10.22　[格雷(Gray)映射]

如果标号是按照格雷映射(即具有最小欧几里得距离的星座点仅差 1 个比特)完成的，请予以校验。对星座点采用格雷映射而不用任意比特映射的目的何在？格雷映射可用于高阶 QAM 调制吗？

题　解

十进制数的二进制表示产生了一种格雷映射。在足够高的 SNR 下，最有可能出错的事件就是检测与实际传输星座点具有最小欧几里得距离的一个星座点。采用格雷映射，这个出错事件产生仅 1 个比特的差错。将单个比特差错赋予比较有可能出错的事件，将多个比特差错赋给不太可能出错的事件，使平均比特差错概率最小。在高阶 QAM 中也可以应用格雷映射。

启动仿真程序。因为在正交分量中的信号看起来像各自的同相分量中的信号，所以对于

无噪声传输在图10.116至图10.125中仅给出了同相分量中的信号。对于有噪声传输，接收端的散布图看起来类似于图10.121。

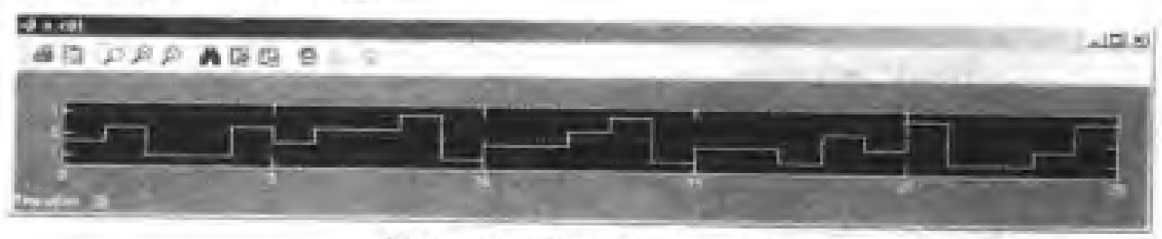


图10.116 同相源信号 $a_c(t)$

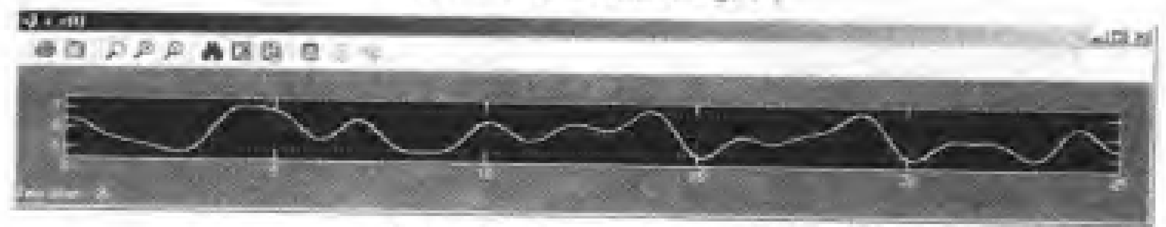


图10.117 同相基带信号 $s_c(t)$

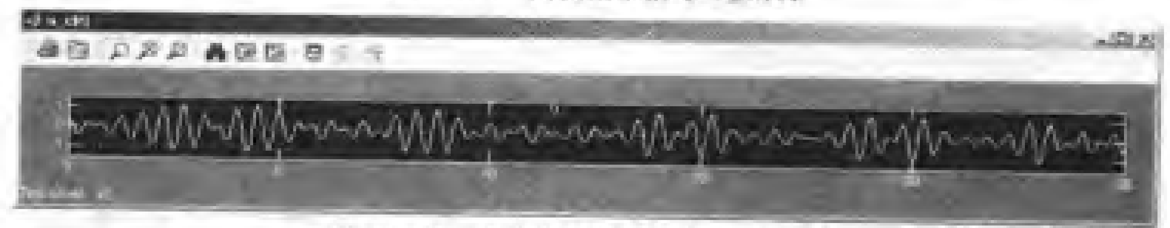


图10.118 发送信号的同相分量 $u_c(t)$

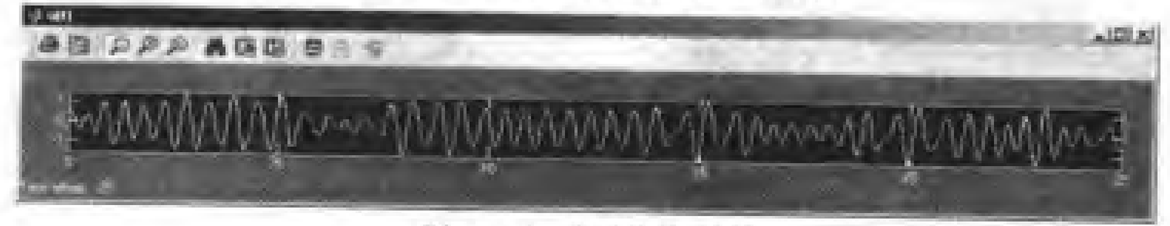


图10.119 发送信号 $u(t)$

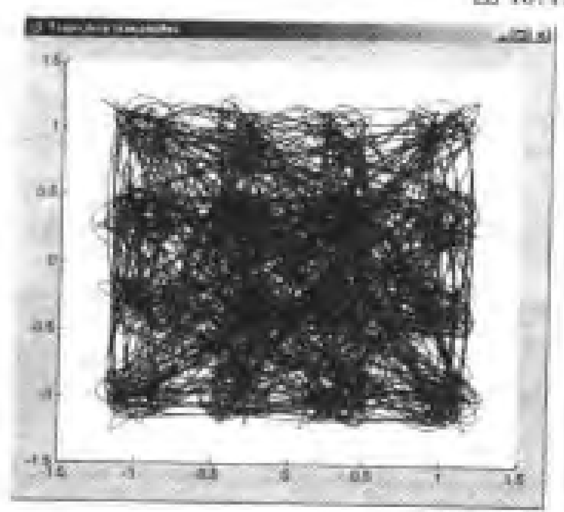


图10.120 采用开根升余弦脉冲的16-QAM发送端信号矢端轨迹图

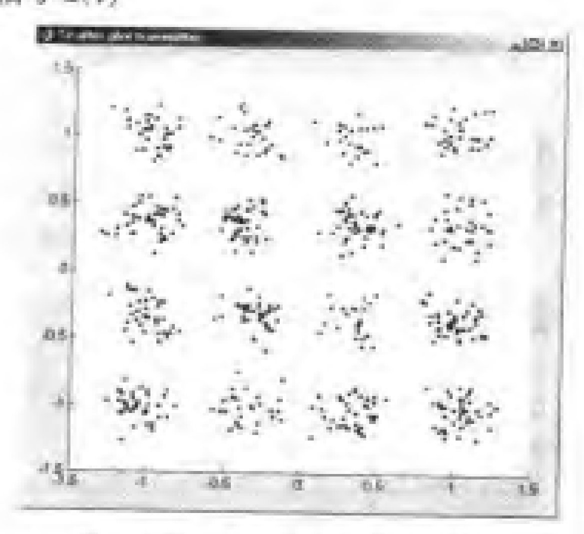


图10.121 采用开根升余弦脉冲的16-QAM发送端散布图

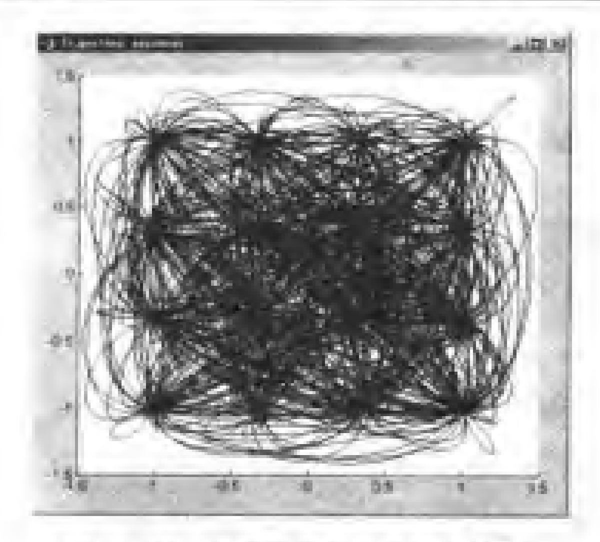


图 10.122 采用开根升余弦脉冲的 16-QAM接收端信号矢端轨迹图

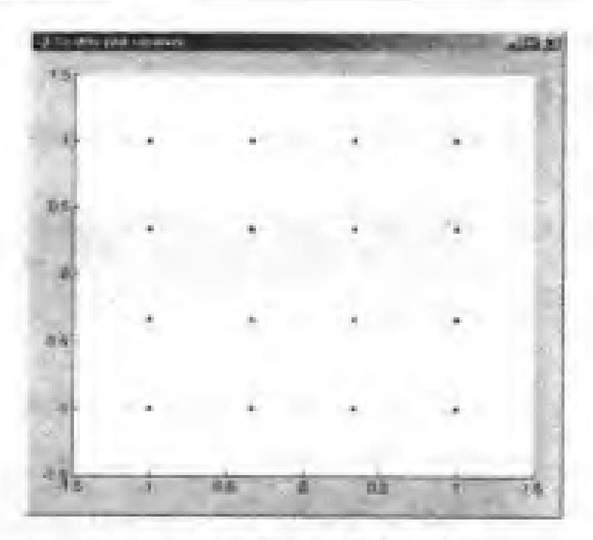


图 10.123 采用开根升余弦脉冲的 16-QAM接收端散布图

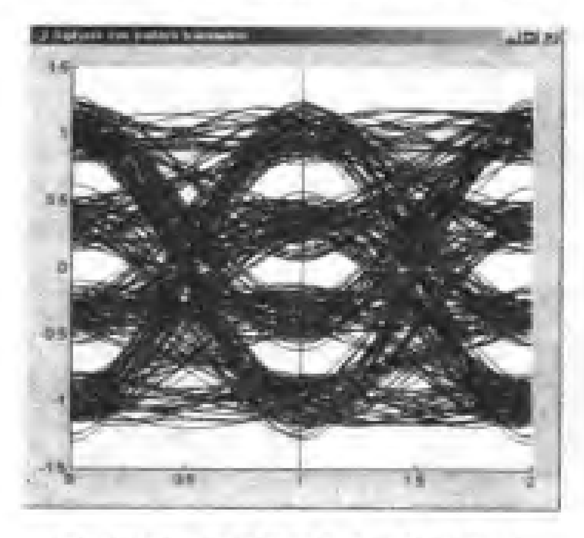


图 10.124 采用开根升余弦脉冲的 16-QAM发送端同相分量眼图

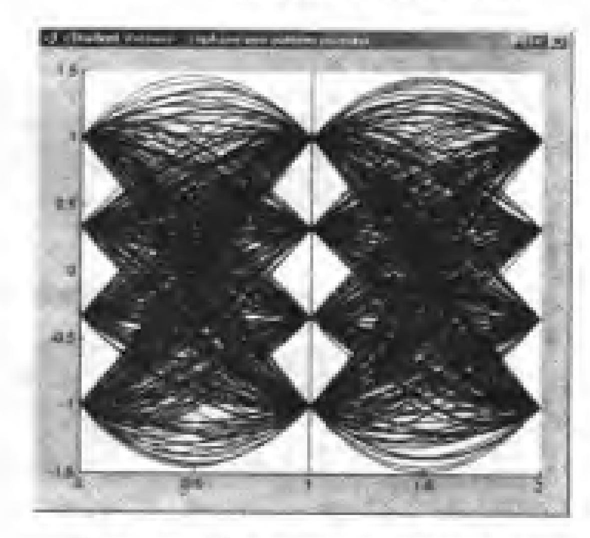


图 10.125 采用开根升余弦脉冲的 16-QAM接收端同相分量眼图

辅导题

辅导题 10.23 [对加性噪声的灵敏度]

对于加性噪声，是 16-QAM 还是 QPSK 更灵敏？利用眼图和信号空间星座图进行解释。

题 解

16-QAM 对加性噪声更灵敏一些。这是因为在相同的符号能量下，星座点的最小欧几里得距离更小一些。在眼图中可以观察到 4 个交叉的结点，这就导致了较小的垂直眼图开启度和对加性噪声的高灵敏度。

10.9.2 采用升余弦脉冲的 16-QAM

现在用升余弦脉冲替换开根升余弦脉冲：

16-QAM > Raised-Cosine

图 10.126 至图 10.129 给出了有关的图形。

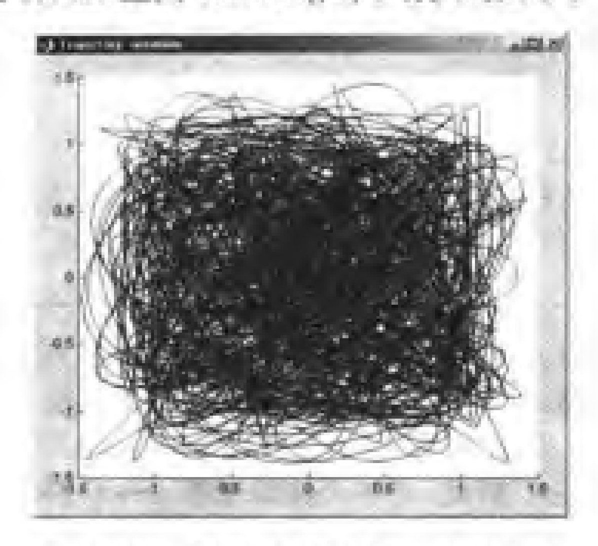


图 10.126　采用升余弦脉冲($\alpha=0.5$)的 16-QAM 接收端信号矢端轨迹图

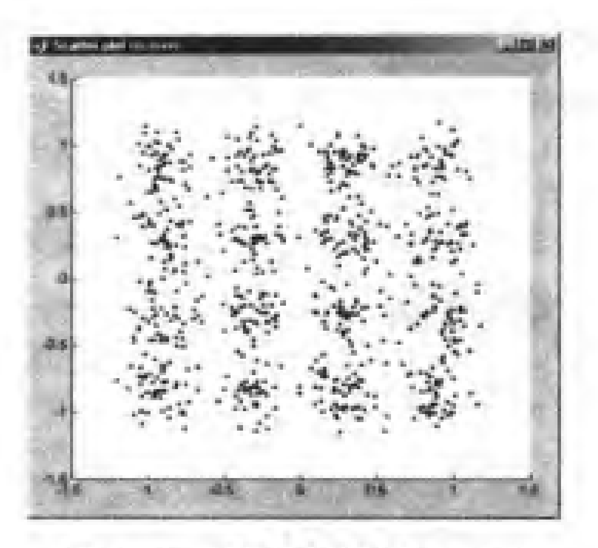


图 10.127　采用升余弦脉冲($\alpha=0.5$)的 16-QAM 接收端散布图

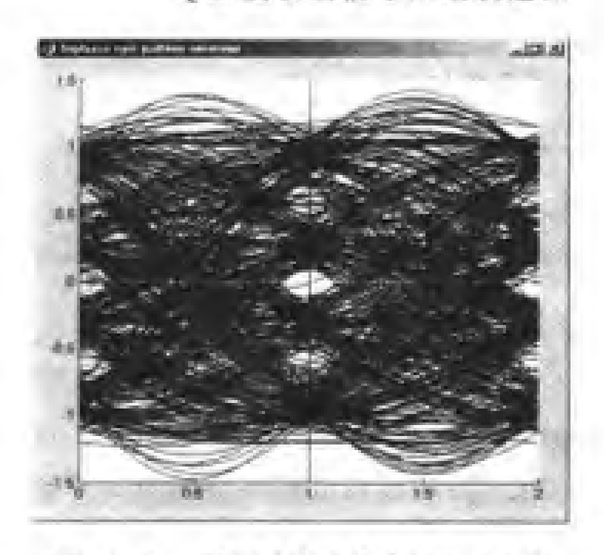


图 10.128　采用升余弦脉冲($\alpha=0.5$)的 16-QAM 接收端同相分量眼图

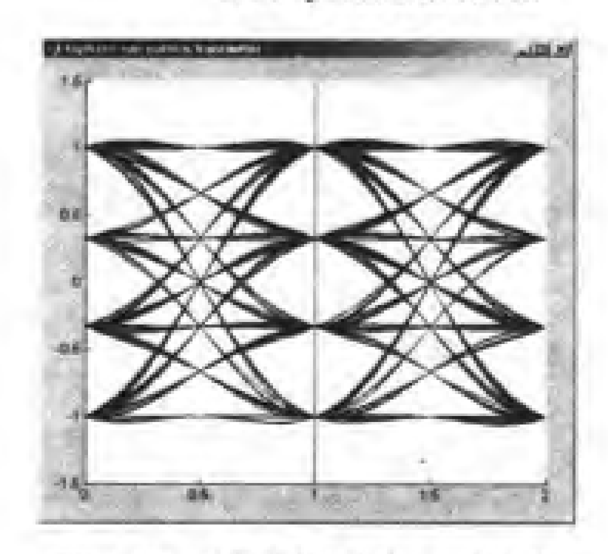


图 10.129　采用升余弦脉冲($\alpha=1$)的 16-QAM 发送端同相分量眼图

辅导题

辅导题 10.24　[采用升余弦脉冲($\alpha=0.5$)的 16-QAM]

如果没有噪声会发生差错吗？利用接收端眼图和散布图给予解释说明。

题　解

利用升余弦发送滤波器并在接收端采用匹配滤波器将产生 ISI。由于信号空间星座点的最小欧氏距离比较小，即使没有噪声，由于错误的滤波器设计所造成的 ISI 也会引起差错。在眼图中可以看出，无差错检测是不可能的。

双击蓝色框 *Roll-off factor*,将其打开后利用滚动条将发送和接收滤波器的滚降系数设定为 $\alpha=0.1$,稍后再设定到 $\alpha=1$。

辅导题

辅导题 10.25 [采用升余弦($\alpha\neq0.5$)的 16-QAM]

1. 如果没有噪声,比 $\alpha=0.5$ 时发生的差错是多了还是少了?

2. 当 $t=0.5T$ 和 $t=1.5T$ 时,对于 $\alpha=1$,为什么在发送端眼图中观察到了结点? 为了解答这个问题,可以使用主菜单中的 *Pulse Shape*。

题 解

1. 对于 $\alpha=0$,升余弦和开根升余弦脉冲是一样的,在接收端有零 ISI。因此,对于小滚降系数 $\alpha\to0$,由于 ISI 产生的 BER 与 $\alpha=0.5$ 相比较是降低了。对于大滚降系数,冲激响应衰减很快。然而,对于下一个相邻符号,ISI 相当强。这一点可以从菜单 *Pulse Shape* > *Convolution* 显示出的两个升余弦脉冲的卷积(如图 10.22 所示)看出。因此,对于大滚降系数 $\alpha\to1$,BER 高于当 $\alpha=0.5$ 时的比特差错率。

2. 对于 $\alpha=1$ 的升余弦脉冲,有:

$$g_{\mathrm{T}}\left(\frac{kT}{2}\right)=\begin{cases}1, & k=0\\ \frac{1}{2}, & k=\pm1\\ 0, & \text{其余 } k\end{cases}$$

因此,在眼图中在$\frac{2k+1}{2}T$处可以观察到结点。

10.10 习题

10.1 如果采用开根升余弦接收滤波器,证明:加性高斯白噪声经采样后仍是白色的。

10.2 BPSK 需要在接收端估计出载波相位。用解析法证明:载波相位偏移 φ 有什么影响,也就是说用 $2\cos(2\pi f_c t+\varphi)$解调。

10.3 在 7.3 节中曾介绍过差分相移键控,它作为一种方法能够在不用做载波相位估计时进行检测。基于图 10.2 所示的 BPSK 模型,构造一个采用 NRZ 矩形脉冲的二元差分相移键控的 Simulink 模型。试观察在载波相位偏移下是否可能有无差错检测。

10.4 用解析法导出在 QPSK 系统中相位偏移 φ 的影响。

10.5 用解析法导出在 QPSK 系统中频率偏移 Δf 的影响。

10.6 避免信号矢端轨迹穿过原点的对角线转移的另一种方法是差分 π/4-QPSK。信息用接续符号的相位差按照如下方式编码:

$$\varphi_k=\varphi_{k+1}+\Delta\varphi_m$$

其中,k 是时间序号,并且

$$\Delta\varphi_m=\frac{\pi}{2}m+\frac{\pi}{4}$$

其中，$m=0,1,2,3$ 是传输符号的序号。用一个例子来证明：从一个 QPSK 星座中每隔一个符号进行提取，而该星座图相对于前面符号的 QPSK 星座相移 $\pi/4$，从而避免了转移通过原点。

10.7 根据图 7.5，如果两个信号波形之间的频率间隔是 $1/(2T)$ 的某个倍数，则可得正交信号波形。选择频率间隔为 $1/(2T)$ 的优点是什么？

10.8 证明：按式(10.8.2)采用余弦脉冲的偏移 QPSK 有恒定不变的包络，假定同相和正交分量之间的延迟为 T（与 MSK 的偏移 QPSK 表示中的情况一样）。

参考文献

[1] J. G. Proakis and M. Salehi. *Communication Systems Engineering*, 2nd ed. Upper Saddle River. N.J.: Prentice-Hall, 2002.

[2] J. Max. "Quantization for minimum distortion," *IRE Transactions on Information Theory*, vol. IT-6, pp. 7–12, March 1960.

[3] J. G. Proakis, *Digital Communications*. 4th ed. New York, N.Y.: McGraw-Hill Inc., 2001.

[4] G. D. Forney. Jr., "Maximum likelihood sequence estimation of digital sequences in the presence of intersymbol interference," *IEEE Transactions on Information Theory*, pp. 363–378, May 1972.

[5] R. Gold, "Optimal binary sequences for spread spectrum multiplexing," *IEEE Transactions on Information Theory*, vol. IT-13, pp. 619–621, October 1967.

[6] R. Gold, "Maximal recursive sequences with 3-valued recursive cross correlation functions," *IEEE Transactions on Information Theory*, vol. IT-14, pp. 154–156, January 1968.

[7] T. Kasami, "Weight distribution formula for some class of cyclic codes," Tech. Rep. R-285, Coordinated Science Laboratory, University of Illinois, Urbana, Ill., April 1966.

[8] R. A. Scholtz, "Optimal CDMA codes," in *National Telecommunication Conference Records* (Washington, D.C.), pp. 54.2.1–54.2.4, November 1979.

[9] D. V. Sarwate and M. B. Pursley, "Crosscorrelation properties of pseudorandom and related sequences," *Proceedings of the IEEE*, vol. 68, pp. 593–619, May 1980.